Characterising the suitability and limitations of metagenomic tools for the detection and discovery of plant viruses

Assessing the limits of detection of viral metagenomic tools

Alisa Prusokiene

Supervisors: Thomas Howard
Neil Boonham
Adrian Fox

School of Natural and Environmental Sciences
Newcastle University

This dissertation is submitted for the degree of
Doctor of Philosophy

Aug 2023

Abstract

Detecting the presence of viral genomes within a plant tissue sample is a vital task, with implications for food security, ecological networks, and biotechnological discovery. Having confidence that a negative is a true negative is necessary, but also difficult - could a novel virus or viroid pass undetected? In the last decade, metagenomic approaches have become widely used for this task, with many diverse methodologies being proposed and becoming implemented. Each is known to have advantages and disadvantages, but the exact boundaries of their limits of detection, and the factors that influence them, have only been studied within restricted subsets of tools. Successful metagenomic detection of viruses and viroids face three main barriers - the divergence of viral genomes, the presence of low-titre viral genomes, and sparsity of taxonomy within our viral reference databases.

In this thesis we develop a benchmarking methodology that allows us to compare a diverse set of approaches in metagenomic viral detection and discovery, and determine the factors that influence their limitations. For this end, we inititally create a novel program for the determination of pairwise substitution distance between short, highly divergent genomes. This program, *Mottle*, is able to successfully quantify substitution distances further than current alternatives. We then use this as a metric, along with other controlled parameters, to determine which software are able to overcome which barriers. We find that there is a trade-off between performance at high divergence and low read depth, with reference sparsity acting as a modulator. Crucially, no approach showed success at detection when all barriers were high. Finally, we apply these tools to previously seen and novel metagenomic datasets, to compare their outputs, and to synthesise conclusions informed by multiple approaches.

I would like to dedicate this thesis to my wonderful partner Lucy, whose mere presence was able to support me through the most difficult times in the course of this study.

Acknowledgements

I would like to acknowledge the guidance and patience of my primary supervisor Thomas Howard, who has gone above and beyond, encouraging my studies through interruptions, extensions, medical issues, and pandemics. I would also like to acknowledge my supervisors Neil Boonham and Adrian Fox, for being calming, positive, and consistently greatly helpful in their guidance. I also acknowledge Aimee Fowkes and Sam McGrieg for their help in obtaining datasets and answering any questions I had. Finally, I would like to acknowledge the whole of Howard Group, both former members and current, for always keeping a lively atmosphere.

Table of Contents

List of Figures

List of Tables

Glossary of Terms

Accuracy • For value prediction tests, how close a set of predictions are to the true values according to some metric e.g. mean absolute error. For classification, the proportion of all tests that predicted the correct class i.e. the proportion of true positives/negatives.

Assembly graph • A data structure generated from reads or k-mers, where each read/k-mer is linked to one, none, or many others based on some similarity metric. See *sequencing read* and *k-mer*

Contig • A nucleic acid sequence generated from joining together a set of smaller, contiguously-overlapping, reads. This overlap between reads may not be perfect, and so a contig may not contain every part of its constituent reads, and is often generated from the consensus of many overlapping reads at each position. See *Sequencing reads*.

***De novo* assembly** • The process of producing one or more Genome assemblies from a set of reads, without the use of targets or templates. See *genome assembly*.

Genome assembly • A set of contigs or scaffolds that have been generated from sequencing reads, whether by *de novo* or targeted assembly, and identified as belonging to a single genome. See *de novo assembly*, *targeted assembly*, and *viral genome*

K-mer • A nucleic acid sequence of a fixed length, K, e.g. 5-mers are composed of five bases. These may be generated from longer sequences, such as reads or genomes. See *sequencing read* and *viral genome*

Nucleic acid sequence • A specific sequence of DNA or RNA bases, that may or may not have a direction, and may exist in molecular form or be stored as information e.g. in a FASTA file

Precision • The proportion of predicted positives that are true positives. See *recall*

Putative viral gene • A putative viral sequence that shows either possible function or shows similarity to sequences that are known to have such function. See *putative viral sequence* and *viral gene*.

Putative viral genome • A set of putative viral sequences that show some sign of co-occurrence and may act together as part of a viral genome, or show similarity to sequences that are known to be part of a viral genome. See *putative viral sequence* and *viral genome*.

Putative viral sequence • A nucleic acid sequence that may be of viral origin, such as by showing similarity to a known viral sequence or appearing on the same sequencing read, contig, or scaffold as a sequence that does. See *sequence similarity*.

Raw read • A nucleic acid sequence, or a sequence of probabilities for each base, as it is generated by a sequencing instrument without further processing. This may be accompanied by a reverse-strand read, and may have additional quality scores generated by the instrument.

Recall • The proportion of all positives that were correctly predicted. See *precision*

Scaffold • A set of ordered contigs that are known to occur on the same molecule but cannot be joined together into a single larger sequence. These usually need extra information to create, outside the original read set, such as optical maps, mate-pair reads, or homologous genomes. See *Contig*

Sequence homology • Similarity or dissimilarity between nucleic acid or proteins sequences due to the presence or absence of shared genetic history. See *Sequence similarity*.

Sequence identity • The proportion of nucleic or amino acids that match between two aligned sequences.

Sequence model • A representation of a set of aligned sequences in some statistical form, such as a Hidden Markov model or Position-specific scoring matrix. This statistical representation can then be queried with an arbitrary sequence to find its similarity to the model.

Sequence similarity • Similarity or dissimilarity between nucleic acid or protein sequences for any reason. This may be in terms of the primary sequence itself or predicted structural similarity of the RNA/Protein molecule produced.

Sequencing read • A nucleic acid sequence, or a sequence of base probabilities, that has been generated with a sequencing instrument. This may have had further processing, such as quality trimming, adapter removal, normalisation, paired-end merging, or correction, but crucially has not been incorporated or merged with other reads to form a contig or graph. This may be accompanied by a reverse-strand read, and may have quality scores generated by the instrument or by computational tools. See *raw read*.

Targeted assembly • The process of producing one or more Genome assemblies from a set of reads, with the use of some kind of target or template, such as seed sequences, sequence models, or target genomes. See *genome assembly*.

Viral gene • A viral sequence that is known to serve a function or is a site for specific interactions e.g. protein coding, ribozyme, structural scaffold etc. See *Viral sequence*

Viral genome • A set of nucleic acid sequences that are known to be of viral origin, which are found together, and act in conjunction to complete the life-cycle of the virus or viroid. May be found in single or multiple separate viral particles.

Viral sequence • Any sequence of nucleic acids that is found within virus or viroid genomes. See *viral genome*.

Chapter 1. Introduction

1.1. Background

Accessing the genomes of microorganisms has traditionally been by culture, directly in a plate or in a host, in a laboratory environment to generate identical, contaminant-free genetic material for DNA or RNA sequencing. This approach was, and still is, a powerful tool in the study of the genome of a single species, but is limited to the microorganisms that can be cultured. This greatly hampers the discovery and monitoring of microorganism communities taken from environmental samples. It has been estimated that >99% of microorganism species found in environmental samples cannot be cultured using conventional techniques (Amann et al., 1995; Hugenholtz et al., 1998).

Advancements in molecular biology and nucleic acid sequencing have resulted in a great shift in the study of microorganisms present in environmental samples, with genetic material being isolated directly from these samples without culture. This process, termed metagenomics by Handelsman et al. (1998), was first used in the mid 1980s to survey microbial diversity in a sample by the comparison of ribosomal RNA segments (Lane et al., 1985; Stahl, 1985; Stahl et al., 1984). These relatively short nucleic acid sequences are conserved within a species, but can distinguish between different species. This approach has allowed the detection, phylogenetic analysis, and monitoring, of many new species of microorganism.

The use of metagenomic techniques to study viral populations has initially lagged behind that of other microorganisms. This is largely because there is no single gene common between all viral genomes equivalent to the rRNA present in many other microorganisms, making the monitoring of their diversity difficult without culture (Edwards and Rohwer, 2005). This difficulty is compounded by the high diversity of viruses within genera, and the rapid rate of evolution of viruses, especially RNA viruses (Holland et al., 1982), which reduces the effectivity of probes on distant species. These difficulties kept viral discovery and monitoring an arduous, time-consuming task as it had to be done by morphological categorisation via specialized electron microscopes and study of serological properties, taking weeks, months, or even years (Anderson et al., 2003) (Figure 1.1). The only way to overcome these difficulties is by directly sequencing the whole metagenome of a sample, a feat that in these early days of metagenomics was not feasible.

Sequencing in the early metagenomic era utilised Sanger sequencing. This relies on amplifying short DNA segments with a randomly inserted radioactively-labelled terminator. At the time, this included a step of cloning these segments into bacterial vectors. While this worked well for

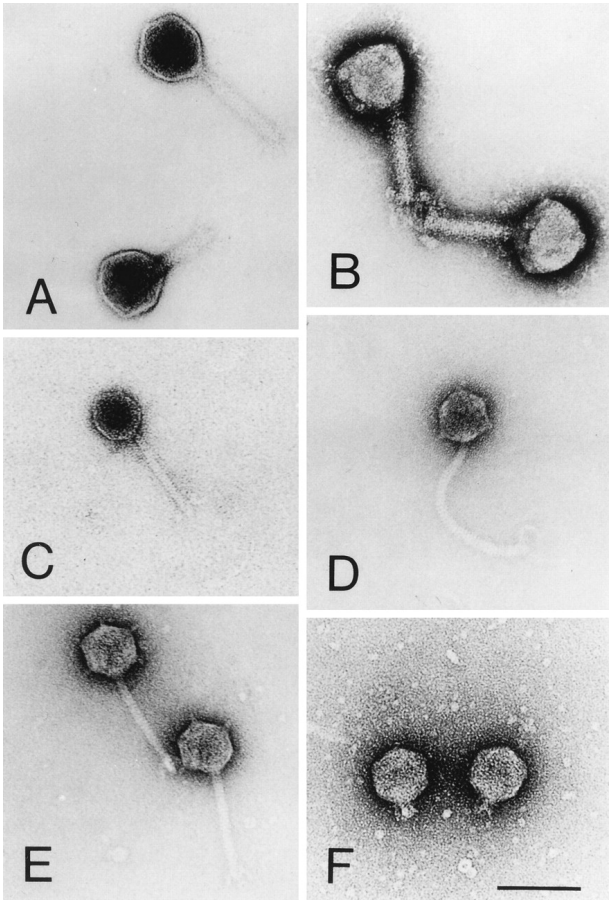


Figure 1.1 Transmission electron microscopy was employed as a common technique to study viral ecology. Shown above are six distinct species of bacteriophages discovered in a marine sample. These phages were categorized into three families and unique morphotypes based only on visual morphology. (A) Myoviridae morphotype 1, exemplified by phage H106/1, with a head lacking antennae and short appendages on the tail. (B) Myoviridae morphotype 2, phage H7/2, with a collar-like structure between the head and tail and short appendages on the tail. (C) Siphoviridae morphotype 1, phage 10-77a, with a head and tail devoid of appendages. (D) Siphoviridae morphotype 2, phage 11 68c, with knob-like appendages on the head and tail and a hook at the end. (E) Siphoviridae morphotype 3, phage H105/1, with knob-like appendages on the head and tail and short appendages. (F) Podoviridae morphotype 1, phage H100/1. Scale bar = 100 nm. Taken from Wichels et al. (1998).

microorganisms, there were problems for viruses: viral genes killing the cloning hosts (Wang et al., 2000), and unclonable modified viral DNA (Warren, 1980). These hurdles were overcome by Breitbart et al. (2002), beginning the field of viral metagenomics, and becoming the first example of whole-genome metagenomics. This was achieved by using random DNA shearing to cut genes into clonable segments, and amplifying fragments by PCR to convert modified bases to their more common counterparts. This gave the first glimpse into the huge diversity of uncharacterised viruses, estimating between 374 and 7,114 viral types in 200 litres of water taken from a marine environment. The sum total of the viral nucleic acid content of a species or ecosystem was termed a 'virome' by Anderson et al. (2003) for the Human Virome Project (Larkin, 2003), retroactively naming the findings of Breitbart et al. a marine virome. Viral metagenomics had a promising beginning, applied to the likes of marine sediment (Breitbart et al., 2002), human stool (Breitbart et al., 2003), and soil (Fierer et al., 2007), but to truly

explore viromes required the processing of many samples. Sanger sequencing was too slow for this, and certainly too expensive.

The great revolution in viral metagenomics came with low cost high-throughput sequencing (Houldcroft et al., 2017), driven by the advancements of next-generation sequencing (NGS) technologies (Hayden, 2014) (Figure 1.2). Though the first of these technologies was developed by Lynx therapeutics, no sequencers were sold to independent laboratories (Brenner et al., 2000). The first major next-generation sequencer to reach the market was Roche 454 in 2005 (Margulies et al., 2005), and was quickly utilised for the exploration of marine viromes (Angly et al., 2006). This, along with ABI SOLiD, Illumina Genome Analyser, Helicos Genetic Analysis System, PacBio SMRT, Ion Torrent, and Oxford Nanopore were responsible for a huge rate of virus discovery, growing the NCBI GenBank virus database by 24.5% between 2009-2010 alone (Benson et al., 2011). While the majority of the virus sequencing effort was geared towards human viruses, especially HIV (Radford et al., 2012), plant virus discovery was greatly accelerated along with these advancements, with over 64 virus discovery papers released in 2009-2011 utilising a variety of these technologies, reviewed in Barba et al. (2014). The viromes of multiple plants has thus been searched, including, but not limited to, sweet potato (Kreuze et al., 2009), grapevine (Coetzee et al., 2010), and pepper (Jo et al., 2017).

Though the technology to sequence plant viromes has matured, there are many questions surrounding implementation: How do you enrich the amount of viral nucleic acid relative to host? How do you store collected tissue for later sequencing without nucleic acid degradation? What method do you use to extract nucleic acids? Which sequencing platform do you use? What bioinformatics pipeline do you choose? How do you detect distantly related viruses? These questions will be explored in the rest this chapter.

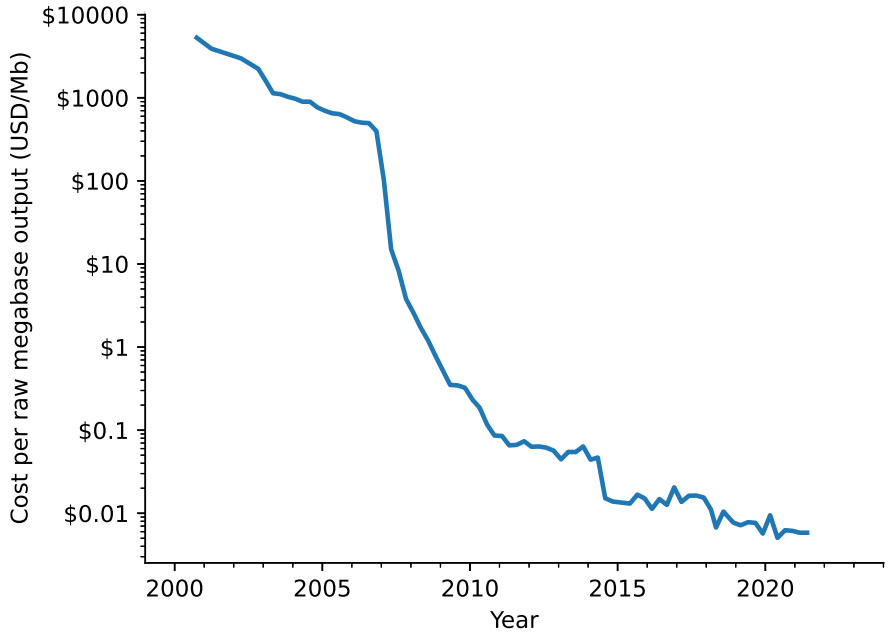
1.2. Methodologies in plant viral metagenomics

Viral metagenomics, including the study of plant viromes, is certainly not a standardised field. From molecular techniques to bioinformatics, it seems that each paper has a novel approach to the problem. Figure 1.3 summarises the main stages of viral metagenomics, each stage containing a variety of approaches that may be used. With such a variety of techniques, how does one decide on which to use?

1.2.1. Collection of samples

In environmental metagenomics, viral particles may be directly collected by sampling the environment, for example taking water or soil samples. In plant virology, viral nucleic acids are locked inside the tissues of their hosts, which must be released prior to requesting. Deciding which tissues to sample is the first step of this process. What tissue contains viruses depends on the method of virus transmission. Viruses may be transmitted vertically, from parent to offspring, by either pollen or directly through seeds (Card et al., 2007; Mims, 1981), leading to much of the plant tissue being infected. Viruses may also be transmitted horizontally, from one plant to

A



B

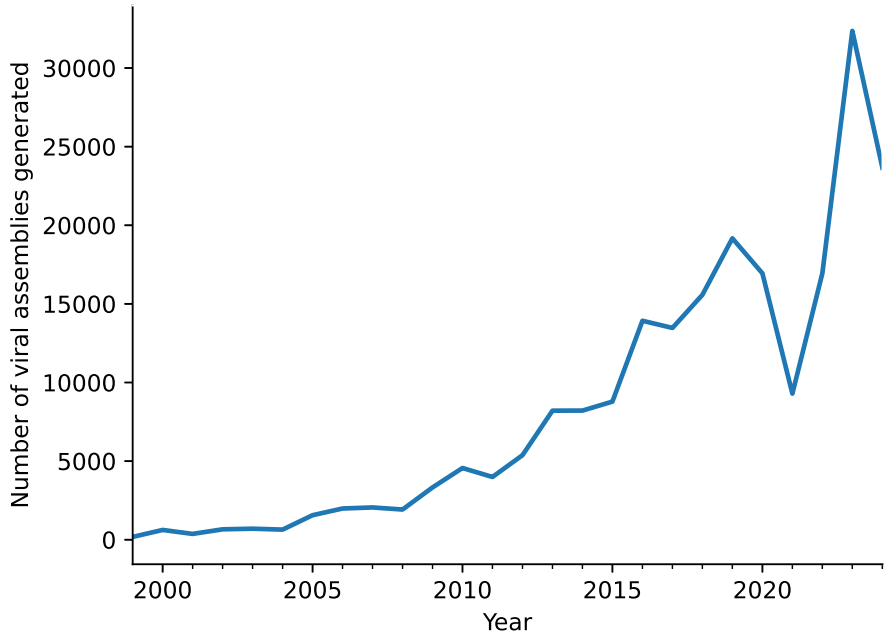


Figure 1.2 A drop in the cost of nucleic acid sequencing has led to an expansion in the study of viral genomes. (A) The cost of sequencing has dropped significantly over time, often out-pacing an exponential decrease. Generated from Wetterstrand (2023). (B) As sequencing costs have dropped, there has been an increasing number of viral genome assemblies generated per year. Generated from National Library of Medicine (2024).

another without parental lineage, by direct contact or through a vector (Blanc et al., 2011; Jia et al., 2018; Lane, 1986). This leads to different tissues being infected depending on the plant, the virus, and its vectors (Brault et al., 2010; Dáder et al., 2017; Dietzgen et al., 2016), thus a bespoke solution is needed for each study. Possible locations include root, stem, leaf, fruit (Jo

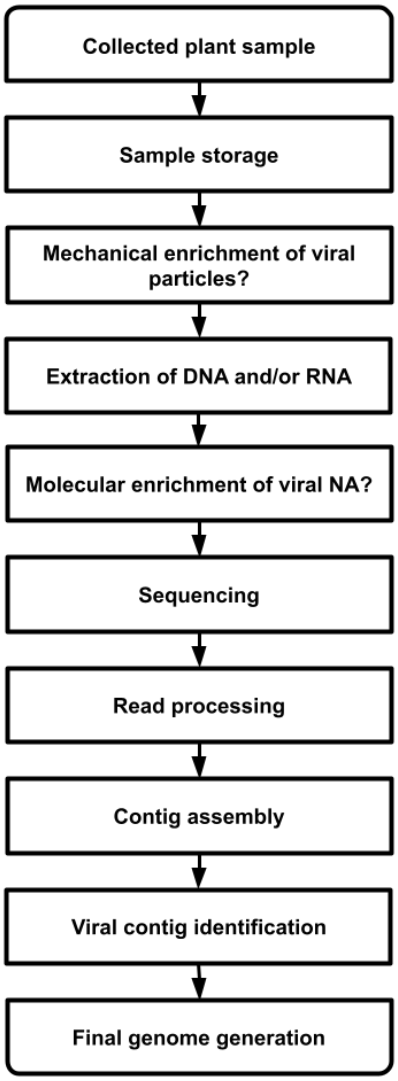


Figure 1.3 An overview of the main stages of viral metagenomics. While most stages are necessary to study a virome, at what stage to enrich viral NA, if at all, can vary. Linking contigs to scaffolds may also be done if a full length genome isn't contained within a contig. Whether to take any confirmatory tests also depends on the study.

et al., 2017), flowers (Schneider et al., 2004), bark (Al Rwahnih et al., 2011), phloem scrapings (Coetzee et al., 2010), or any combination thereof.

1.2.2. Sample storage

Once samples have been collected, they must be preserved until extraction. This can be a matter of hours or years, with viruses being sequenced from 750 and 1000 years old tissue (Adams et al., 2018; Peyambari et al., 2019). These tissues were naturally preserved in an arid environment, but the problem of how to store tissues in a laboratory setting is one with many solutions. This is especially important as >90% of plant viruses are RNA viruses (Waterhouse et al., 2001), with unprotected RNA being highly susceptible to degradation (Eigner et al., 1961). Some proposed solutions are listed in Table 1.1. The most common procedures are to freeze, dry/dehydrate, or freeze-dry the tissue (Gould, 1999), with the latter being especially powerful, with RNA virus genomes being successfully recovered after over 30 years of storage (Adams

et al., 2018, 2017). Though there has been study into the effectiveness of preservation procedures on single viral species, there has not been quantification of what kind of biases these may have towards the differential degradation of viral species, which may bias metagenomic studies.

Table 1.1 Methods of preserving viral RNA for later extraction.

Procedure	Method	Storage conditions
Freezing	Drop tissue in liquid nitrogen or place in freezer	-20°C/-80°C
Drying	Air-dry in warm conditions with low humidity, possibly with a dessicant, or briefly dry at 60°C	RT with dessicant
Chemical dehydration	Refrigerate tissue in presence of Anhydrous Calcium Chloride	4°Cin sealed container with dessicant
Lyophilisation	Freeze-dry tissue	RT in dark
FTA PlantSaver	Press tissue into card	RT with dessicant
RNAlater	Place tissue in solution	-20°C/-80°C
Ethanol	Extract RNA and place in 70%-100% Ethanol solution	4°Cor lower
RNAstable	Extract RNA and place in solution	4°Cor lower
Create cDNA	Extract RNA and use in reverse- transcriptase reaction to create cDNA	Store as lab would DNA extractions

For each procedure, possible methods of implementation are listed, as well as the conditions of storage after treatment. RT = Room Temperature.

1.2.3. Enrichment of viral nucleic acids

Viral nucleic acids are a small fraction of the total nucleic acid of a plant (Roossinck et al., 2015). Directly sequencing the total DNA or RNA of a plant would therefore have a low read depth of viral nucleic acids, possibly missing low-titre viruses entirely. Enrichment of viral nucleic acids is thus essential for the study of plant viromes. A variety of solutions have been proposed, each with their own drawbacks (Table 1.2). Filtering viruses based on size, such as by centrifugation, and digestion of non-encapsidated nucleic acids are techniques that may be performed before the extraction of nucleic acids, while the others are done after extraction. These methods may also be performed in combination (Figure 1.4). Choosing a method of enrichment depends on what kind of viruses you are looking for, DNA/RNA, encapsidated, persistent, or from a known family.

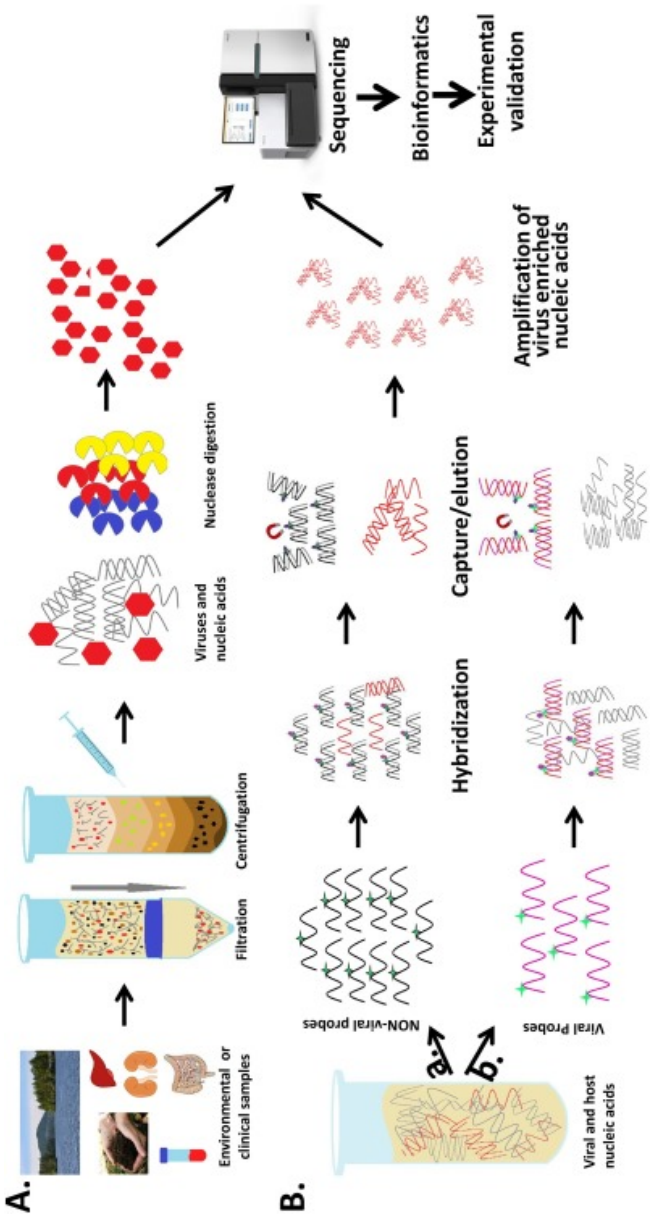


Figure 1.4 Examples of virus enrichment methodologies. (A) Positive selection of viral particles. Samples are filtered or centrifuged to select for virus-like particles. This is then nuclease treated to remove nucleic acids that also passed through, then the nucleic acids within the VLPs are released for sequencing. (B) Negative selection against non-viral nucleic acids and positive selection for viral nucleic acids using probes. Viral nucleic acids can be further amplified by PCR before sequencing. Taken from Kumar et al. (2017).

Table 1.2 Methods of enriching viral nucleic acids.

Method	Drawbacks
Size filtration or density-based enrichment	Only works for viruses of specific morphology (Segura et al., 2011), possibly missing large viruses (Parras-Moltó et al., 2018)
Precipitation with Additives	Low recovery of labile virus particles (Segura et al., 2011)
Chromatography	May miss viruses that are not a specific size or have specific surface properties (Segura et al., 2011)
Nuclease digestion	Biased towards viruses that spend more time encapsidated, and entirely misses non-encapsidated viruses (Roossinck et al., 2015)
Random amplification	Bias towards circular genomes and against areas of extreme GC content(Kim and Bae, 2011; Parras-Moltó et al., 2018)
dsRNA enrichment	Misses transcribed DNA virus genes and sRNA viruses that do not have a dsRNA intermediate (Roossinck et al., 2015)
siRNA enrichment	May miss persistent viruses that suppress or don't trigger RNA silencing (Roossinck et al., 2015)
Subtractive hybridization	Removes asymptomatic and persistent viruses that are present in the control (Roossinck et al., 2015)
rRNA depletion	Misses DNA virus sequences that are not transcribed, and still has a relatively small yield of viral RNA (Pecman et al., 2017)
Viral probes or PCR enrichment	Can only be used for known families, and may miss distant species with little to no homology (Houldcroft et al., 2017)

1.2.4. Nucleic acid extraction

The nucleic acids stored within plant tissue need to be accessed for their sequencing. The tissue is first broken down to release the contents contained within the cell wall or viral capsid, either through the use of a bead beater, sonication, a grinding machine, or through the use of snap-freezing with liquid nitrogen with a pestle and mortar to manually grind the tissue.

Vortexing with phenol is particularly effective for this task. Once broken up, cell membranes may need to be disrupted. Detergents such as sodium dodecyl sulfate (SDS) or cetrimonium bromide (CTAB) work well for this, with both also denaturing proteins that might break down or modify nucleic acids, and CTAB being especially effective for plant tissues (Azmat et al., 2012). This leaves nucleic acids in solution, along with cell fragments and denatured proteins.

To finally extract these nucleic acids they may either be seperated in solution or adsorbed to silica. Phenol-chloroform extraction is one such method, separating nucleic acids from other molecules. Acid guanidinium thiocyanate-phenol-chloroform (AGPC) extraction is a technique that is widely used for RNA, as it can perform membrane disruption and RNA separation in a single step. nucleic acids are known to bind to silica under certain circumstances. This has been exploited in the boom method, which utilises the binding of nucleic acids to silica beads which can then be removed from the solution and washed to release the nucleic acids. The use of magnetic beads can allow the automation of this technique for higher-throughput. Many commercial kits instead use a spin-column containing a silica gel membrane, which only the nucleic acids can bind to. Once RNA, DNA, or both are extracted, they must be further prepared for sequencing. This is done according to the sequencing platform that will be used.

1.2.5. Sequencing

Sequencing technologies can be broadly grouped into three categories: first-, second-, and third-generation (Table 1.3). The first generation, Sanger sequencing, is an accurate but expensive and low throughput technique. These properties make it unsuitable for large virome studies, which would require high amounts of labour and cost. The second generation of sequencers are much more suitable, generating short reads but with very high throughput (Figure 1.5). The reads originally produced by Illumina platforms, with a length of 75bp or less, were appropriate for reference-guided assembly, but were not very reliable for de novo assembly. Though not as significant of a challenge for viral metagenomics, this did prevent the technology being used for the creation of high-quality draft viral genomes. 150bp reads allowed higher quality assemblies, but the greatest advancement for genome assembly was paired-end, and later mate-pair reads. This involved sequencing both ends of a fragment a certain length apart, allowing these reads to be linked as being part of the same molecule, greatly increasing the ability to assemble long sequences, and the confidence of viral assemblies.

Table 1.3 Description of sequencing technologies and the generation they are part of.

Platform	Gen.	Description
Sanger sequencing	1st	A low-throughput technique that involved the plasmid cloning of fragmented or amplified nucleic acids, and the use of radioactive or fluorescently tagged dideoxy terminators for sequencing. Though this was automated and parallelised in machines, the length of time needed to sequence even the small viral genomes and the associated cost led to it being supplanted by later generations.
454 / pyrosequencing	2nd	The first non-sanger sequencing technique used in viral metagenomics. This utilises a sequencing-by-synthesis methodology, where DNA molecules are amplified through emulsion PCR. When pyrophosphates are released during DNA synthesis, they are enzymatically converted to a light signal which is detected by a camera. This was a much cheaper and higher throughput technique to Sanger Sequencing, but had problems with homopolymeric runs and errors due to artificial amplification (Kumar et al., 2017).
Solexa / Illumina	2nd	This further reduced sequencing costs greatly. Originally developed by Solexa, but later bought by Illumina, this is another sequencing-by-synthesis technique, using reversible dye-terminators that enable the identification of bases as they are incorporated into sequence clusters. While a single run can be expensive, this technique is very high throughput as many sequences can be incorporated into a single chip. This allows a large sequencing depth while multiplexing many samples, giving the technique dominance in viral metagenomics.
Ion torrent	2nd	An alternative to pyrosequencing and Illumina technologies. Sequences are prepared by emulsion PCR, similar to 454, and also detect a product of nucleotide incorporation in DNA synthesis. Instead of pyrophosphates, though, this detects protons that are released using an ion sensor. This detection method is much faster than pyrosequencing and has largely superseded it. The smaller out than Illumina sequencing make this technique less useful for virome studies.

Table 1.3 Description of sequencing technologies and the generation they are part of (cont.)

Platform	Gen.	Description
PacBio	3rd	A single-molecule real-time sequencing platform, this technique is used to generate long reads without amplification, which can introduce bias. A strand-displacing DNA polymerase is affixed within small chambers, which incorporates fluorescently-tagged nucleotides to a single DNA molecule. This incorporation cleaves and activates the tag, which is captured by camera. The small well size, on the zeptoliter scale, greatly reduces noise and allows continuous synthesis. The long reads allow capturing the whole of a viral genome in a single read, but its high cost, low output and high error rates prevent wide adoption for virome study.
Nanopore	3rd	A long-read, single-molecule technology characterised by the use of small pores, just large enough to fit a nucleotide through. An immobilised motor enzyme pushes a nucleic acid molecule through the pore. As the molecule moves through, the differently shaped bases alter the current running through the pore in a characteristic way, allowing its sequencing. This technology has been miniaturized to a handheld device, which is cheaper than PacBio, and has a quick run-time of a few hours. Despite high adoption, the low output and high error rate make it unsuitable for viral metagenomics.

Comparisons of Sanger, Pyro-, and Illumina sequencing for metagenomics has shown that for a simple community (10 genomes) all sequencing technologies assembled to a similar quality. A complex community (100 genomes), though, was shown to produce the best assemblies when sequenced by Illumina platforms, especially with paired reads (Mende et al., 2012).
Third-generation sequencing, those that produce long reads, have only recently been taken up for viral metagenomics. Despite the ability to sequence a whole viral genome in a single read, where multiple reads can find sequence and structural variants, a low read number hampers Oxford Nanopore sequencing's suitability for detection of early, low titre viral infections. Additionally, the high error rate of its long reads means that a consensus of multiple reads must be taken to produce accurate genomes (Sun et al., 2022). Regardless of sequencing platform, a large problem is contamination. Many viruses found in virome studies may actually originate from laboratory components (Asplund et al., 2019). Instead of which platform to use, thought needs to be put into keeping sterile conditions during sequencing preparation to prevent this contamination.

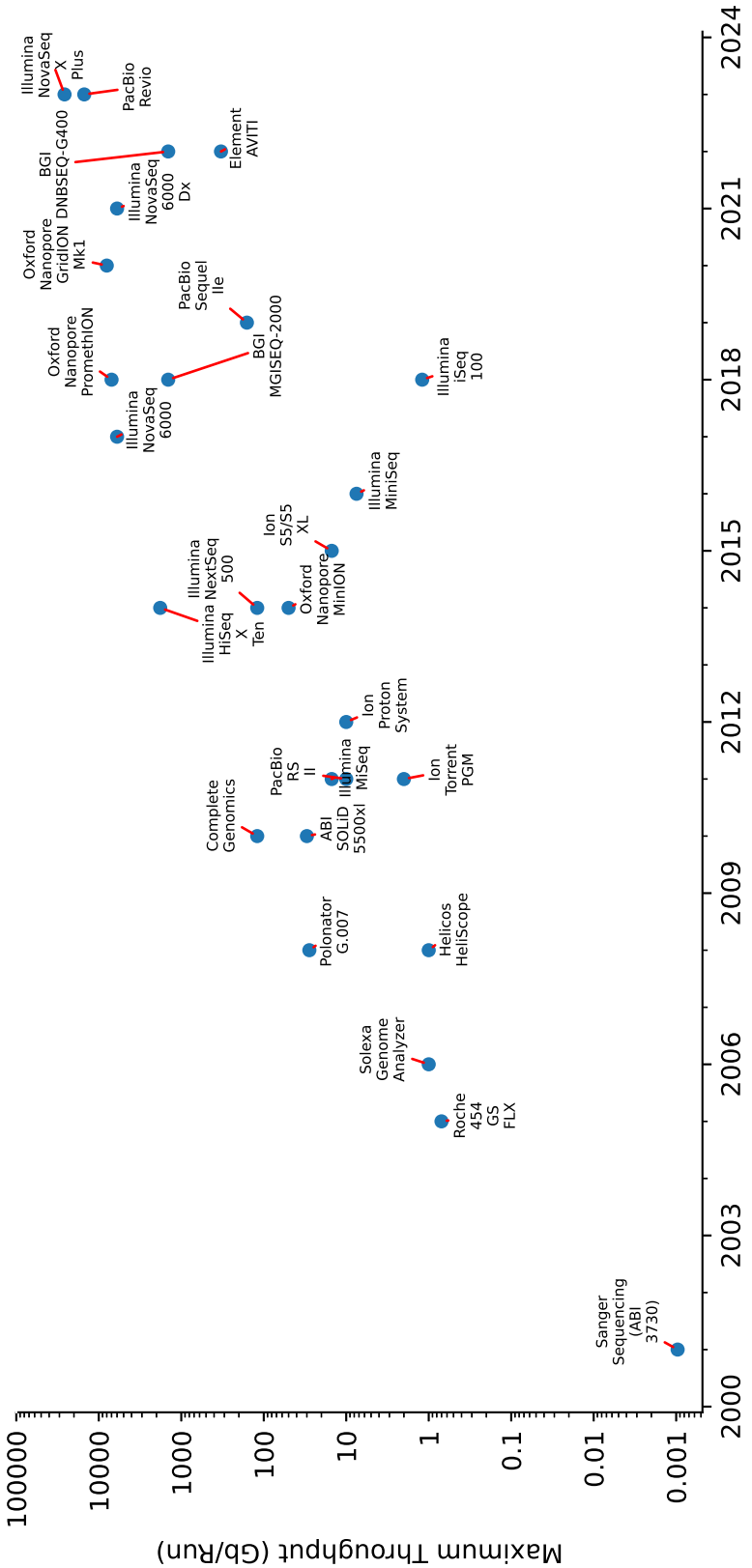


Figure 1.5 The increasing of output from sequencing technologies over time. These high throughput techniques have given lower sequencing costs, as well as the high coverage needed for virome studies. Maximum throughput refers to the maximum raw sequence output in gigabases per run according to the manufacturer.

1.2.6. Read pre-processing

Reads from sequencing may need to be processed to allow high-quality assembly. Errors and artefacts appear in every sequencing platform (Wan et al., 2015), with high-coverage Illumina sequencing having on average an error in some read at each base of a genome (Marçais et al., 2015). Filtering and quality trimming reads, as well as removing adapters or primers introduced in sequencing, is the first step in any bioinformatic process. In one study, despite quality trimming removing 13-16% of reads and 24-27% of base pairs, the quality and coverage of assembly was greatly improved, with a 2-fold increase of reads mapping back to their original genome (Mende et al., 2012). Tools for this are summarised in Table 1.4. There are three main approaches: utilising phred scores, aligning reads, evaluating k-mers. Phred scores represent the sequencing quality of each base in a read, in an experiment independent manner. Trimming bases with a low quality score, whether with a direct threshold, or in an adaptive manner, is the simplest way of processing reads. Unfortunately much information can be lost in these bases, some of which may be legitimate and vital for high-quality assembly. Another method to process reads is by aligning them to each other. Regions that are aligned between many reads must be legitimate, and single base errors become obvious. Similarly, breaking reads down to k-mers, all the subsequences of a certain length that make up the read, can be used to find legitimate bases. If a k-mer is abundant in a sequencing run, then it is likely to be legitimate and the regions of reads that contain it may be kept. Less abundant k-mers must then be analysed to find whether they represent sequence variation, or are simply errors.

Table 1.4 Quality trimming and/or adapter removal tools used for read-preprocessing.

Trimming tool	Quality measure	Removes adapters / primers?
SeqTrim (Falgueras et al., 2010)	Phred score	Yes
Quake (Kelley et al., 2010)	Trusted k-mers	No
Cutadapt (Martin, 2011)	N/A	Yes
Sickle (Joshi and N, 2011)	Phred score	Yes
Btrim (Kong, 2011)	Phred score	Yes
Coral (Salmela and Schröder, 2011)	Read alignment	No
AlienTrimmer (Criscuolo and Brisse, 2013)	N/A	Yes
QC-Chain (Zhou et al., 2013)	Phred score	Yes
SEECER (Le et al., 2013)	Read alignment	No
Trimmomatic (Bolger et al., 2014)	Phred score	Yes

Table 1.4 Quality trimming and/or adapter removal tools used for read-preprocessing. (cont.)

Trimming tool	Quality measure	Removes adapters / primers?
QTrim (Shrestha et al., 2014)	Phred score	No
ngsShoRT (Chen et al., 2014)	Phred score	Yes
Blue (Greenfield et al., 2014)	Trusted k-mers	No
LoRDEC (Salmela and Rivals, 2014)	Trusted k-mers	No
Fiona (Schulz et al., 2014)	Read alignment	No
QuorUM (Marçais et al., 2015)	Trusted k-mers	No
PEAT (Li et al., 2015)	N/A	Yes
ACE (Sheikhizadeh and Ridder, 2015)	Trusted k-mers	No
BFC (Li, 2015)	Trusted k-mers	No
Karect (Allam et al., 2015)	Read alignment	No
NxTrim (O'Connell et al., 2015)	N/A	Yes
UrQt (Modolo and Lerat, 2015)	Phred score	No
Rcorrector (Song and Florea, 2015)	Trusted k-mers	No
Reptile (Pal and Aluru, 2015)	Trusted k-mers	No
SeqPurge (Sturm et al., 2016)	N/A	Yes
AdapterRemoval v2 (Schubert et al., 2016)	N/A	Yes
LoRMA (Salmela et al., 2017)	Trusted k-mers	No
HALC (Bao and Lan, 2017)	Read alignment	No
IterativeErrorCorrection (Sameith et al., 2017)	Trusted k-mers and read alignment	No
FMOE (Huang and Huang, 2017)	Read alignment	No
Atropos (Didion et al., 2017)	N/A	Yes
cutPrimers (Kechin et al., 2017)	N/A	Yes
Fastp (Chen et al., 2018a)	Phred score	Yes
FastqPuri (Pérez-Rubio et al., 2019)	Phred score	Yes
FLAS (Bao et al., 2019)	Read alignment	No

Another problem that assemblers may face is uneven read depth (Figure 1.6). Read depth varies throughout a genome, especially within RNA viruses where coverage within a single run can vary from tens to tens of thousands. This may be due to areas with high G+C content or secondary structures (Wan et al., 2015). To correct this before sequencing, normalisation techniques can be applied. DigiNorm, an early implementation, does this by analysing the makeup of reads, where reads that contain redundant information are removed. NeatFreq (McCorrison et al., 2014), BigNorm (Wedemeyer et al., 2017), and ORNA (Durai and Schulz, 2019) improves on this by incorporating base quality information and including deeper analyses of read redundancy. This also has the side effect of increasing the speed of assembly.

1.2.7. Contig assembly

Whether sequencing reads are long or short, they need to be joined together to create longer sequences and/or correct for read errors. This process is called assembly, and its output is a set of contigs that each represent the consensus of a set of reads. There are two broad categories of assemblers: overlap based approaches and de Bruijn graphs (Paszkiewicz and Studholme, 2010). In overlap based approaches, reads are aligned to each other to find overlaps between them. These reads can either be joined by a greedy algorithm, that incrementally merges the most overlapping reads/contigs together, or a graph based approach which represents each read as a node and overlap as an edge, with assembly being a traversal of this graph. de Bruijn graphs utilise the breaking down of reads to k-mers, all the subsequences of a certain length (called k) that make up the read, to form the nodes of a graph, with edges representing shared k-1-mers. The choice of k is an important parameter in this approach, where longer k-mers produce longer contigs, but have more errors (Wan et al., 2015). As k-mers are short, and as similar reads share a majority of k-mers, de Bruijn graphs can be much faster to construct. The downside, though, is that k-mers lose some of the connection information of reads, and may merge repeats or parts of different reads together, producing more complex graphs with shorter contigs (Wan et al., 2015; Yoon et al., 2018).

Regardless of the method of construction, assembly graphs can be very difficult to resolve, especially when containing multiple similar genomes (Figure 1.7). Many tools for the assembly of metagenomes have been developed to solve this problem (Table 1.5). Notable approaches include: MetaCAA, which clusters sequences to speed up assembly, with further assembly to connect missed reads. SPA and SFA-SPA use a gene finder to identify partial protein sequences and assembles them into full length proteins. PRICE, IVA, IRMA, and MATAM use a nucleic acid sequence database to guide assembly, while GeneStitch, IDBA-MTP, and MEGAN use a protein database. GenSeed-HMM, Xander, and MegaGTA similarly use hidden markov models (HMMs) to guide their assembly. Cortex and VARI attach metadata to nodes and edges by tagging them with 'colours'. Oases, IDBA-tran, VirAmp, and TraRECo utilise various sizes of k-mers for more robust assembly. BinPacker and Shannon utilise the difference in coverage of reads to separate ambiguous nodes in de Bruijn graphs into multiple contigs. MetaVelvel-SL does this instead by using machine learning techniques to identify hybrid nodes. Trinity and

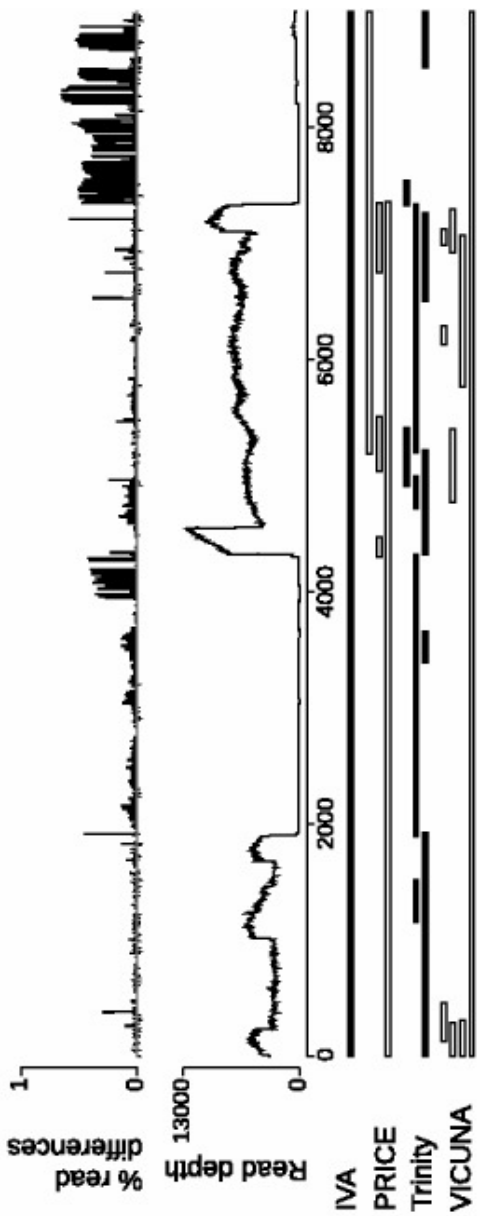


Figure 1.6 Uneven read depth and high sequence diversity in certain regions of HIV-1. The proportion of single base difference per read (top), the depth of each read (middle), and the contigs constructed for this region from various assembly tools used for viral metagenomics (bottom). The problems of reads depth and sequence divergence, if not handled properly, can produce short and fragmented assemblies, with some areas of the genome being missed. Taken from Hunt et al. (2015).

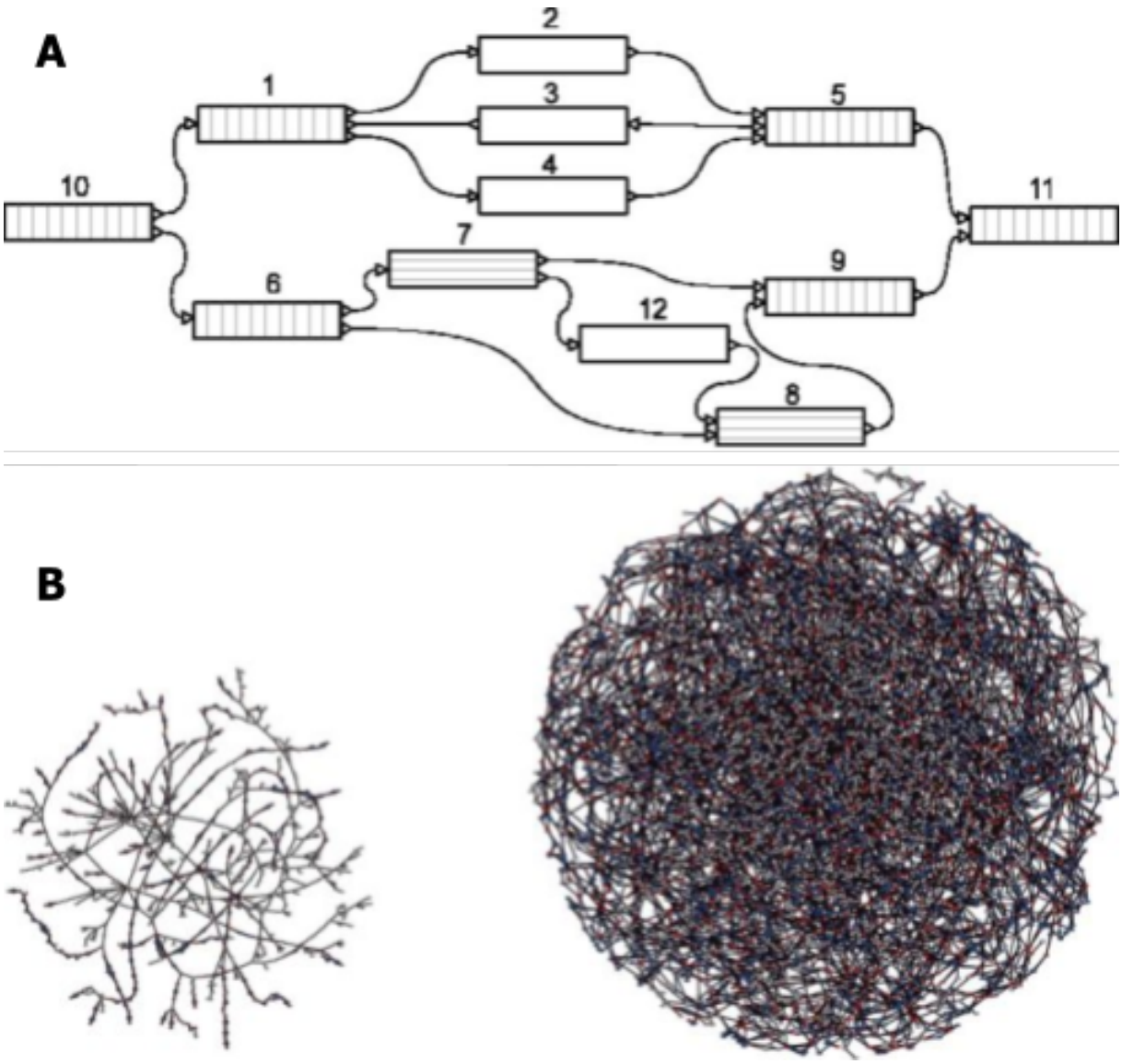


Figure 1.7 Assembly graphs of metagenomes may be ambiguous and complex. (A) A simple contig graph, which has linear overlapping reads/k-mers merged to contigs, but ambiguous connections prevent this from being further resolved to longer contigs. These ambiguities can come from sequence variation, the merging of repeats to a single node, or erroneous connections between similar regions in the same or different genomes. If not properly resolved this can lead to very short contigs, or chimeric contigs that merge different genomes. (B) Graph complexity greatly increases with higher numbers of similar genomes. Largest connected components of the contig graphs for two simulated sequencing datasets for three (left) and seven (right) *Escherichia coli* genomes. Adapted from Nijkamp et al. (2013).

rnaSPAdes detect bubbles in de Bruijn graphs to find RNA isomorphs, while Cortex and MaryGold use this to find all sequence variants. Focus utilises a hybrid graph, which merges the nodes of a complex assembly graph at multiple levels to group sequence variants and resolve ambiguities. The most notable for plant virome studies are VICUNA, PRICE, IVA, VirAmp, IRMA, and SAVAGE, which are optimised for the assembly of viral genomes, taking into account their short lengths and high sequence variation. Regardless of the tool used, an essential step for obtaining high quality assemblies is parameter tuning to match the properties of the genome (Wan et al., 2015).

Table 1.5 *De novo* assemblers used in viromics.

Assembler	Algorithm	Target
Trans-ABySS (Robertson et al., 2010)	de Bruijn	Metatranscriptome
Genovo (Laserson et al., 2011)	Overlap	Metagenome
Meta-IDBA (Peng et al., 2011)	de Bruijn	Metagenome
Trinity (Grabherr et al., 2011)	de Bruijn	Metatranscriptome
Metavelvet (Namiki et al., 2012)	de Bruijn	Metagenome
Ray Meta (Boisvert et al., 2012)	de Bruijn	Metagenome
VICUNA (Yang et al., 2012)	de Bruijn	Virome
IDBA-UD (Peng, 2012)	de Bruijn	Metagenome
MAP (Lai et al., 2012)	Overlap	Metagenome
Cortex (Iqbal et al., 2012)	de Bruijn	Metagenome
Oases (Schulz et al., 2012)	de Bruijn	Metatranscriptome
GeneStitch (Wu et al., 2012)	de Bruijn	Metagenome
SPA (Yang and Yooseph, 2013)	de Bruijn	Protein-coding genes
IDBA-MT (Leung et al., 2013)	de Bruijn	Metatranscriptome
PRICE (Ruby et al., 2013)	de Bruijn	Virome
Xgenovo (Afiahayati and Sakakibara, 2013)	de Bruijn	Metagenome
IDBA-tran (Peng et al., 2013)	de Bruijn	Metatranscriptome
Omega (Haider et al., 2014)	Overlap	Metagenome
MetaCAA (Reddy et al., 2014)	Overlap	Metagenome
SOAPdenovo-Trans (Xie et al., 2014)	de Bruijn	Metatranscriptome
VirAmp ((Wan et al., 2015)	Either	Virome
SFA-SPA (Yang et al., 2015)	de Bruijn	Protein-coding genes
IDBA-MTP (Leung et al., 2015)	de Bruijn	Metatranscriptome
MetaVelvet-SL (Afiahayati, 2015)	de Bruijn	Metagenome
IVA (Hunt et al., 2015)	de Bruijn	Virome
Xander (Wang et al., 2015)	de Bruijn	Protein-coding genes
Bridger (Chang et al., 2015)	de Bruijn	Metatranscriptome
MEGAHIT (Li et al., 2016)	de Bruijn	Metagenome

Table 1.5 *De novo* assemblers used in viromics (cont.)

Assembler	Algorithm	Target
Focus (Warnke-Sommer and Ali, 2016)	Overlap	Metagenome
IRMA (Shepard et al., 2016)	Either	Virome
BinPacker (Liu et al., 2016b)	de Bruijn	Metatranscriptome
Shannon (Kannan et al., 2016)	de Bruijn	Metatranscriptome
BASE (Liu et al., 2016a)	Overlap	Metagenome
MegaGTA (Li et al., 2017)	de Bruijn	Metagenome
MetaSPAdes (Nurk et al., 2017)	de Bruijn	Metagenome
VARI (Muggli et al., 2017)	de Bruijn	Metagenome
MEGAN (Huson et al., 2017)	de Bruijn	Metagenome
SAVAGE (Baaijens et al., 2017)	Overlap	Virome
GenSeed-HMM (Alves et al., 2016)	Either	Metagenome
MATAM (Pericard et al., 2018)	Overlap	Metagenome
TraRECo (Yoon et al., 2018)	de Bruijn	Metagenome
rnaSPAdes (Bushmanova et al., 2018)	de Bruijn	Metatranscriptome

1.2.8. Identification of viral contigs

Once you have a collection of contigs, the problem becomes to find which is of viral origin. It is a relatively simple task to filter out known contigs, such as host sequences and known viruses, using standard homology search tools like BLASTx (Camacho et al., 2009). This, though, still leaves a large portion of unknown contigs, in some studies even exceeding the number of known contigs (Roossinck, 2012). There is no single conserved sequence between all viruses, and even relatively conserved sequences in families, such as polymerases, can be so distant as to barely be recognised (Bolduc et al., 2012; Roossinck, 2012). Viruses, especially RNA viruses that encode their own replication enzymes which commonly infect plants, have extremely high mutation rates (Holland et al., 1982; Steinhauer and Holland, 1987), leading some to propose that RNA viruses exist as a diverse quasispecies (Lauring and Andino, 2010) (Figure 1.8). When an individual virion that is in the outer edge of the quasispecies infects a new host, it becomes the new centre of the local quasispecies, so the diversity of viruses varies between individual hosts (Desbiez et al., 2011). This high divergence from known sequences prevent some viruses being identified by simple homology searches. While some have created more sophisticated homology search tools, others have proposed methods that can identify novel viruses in a

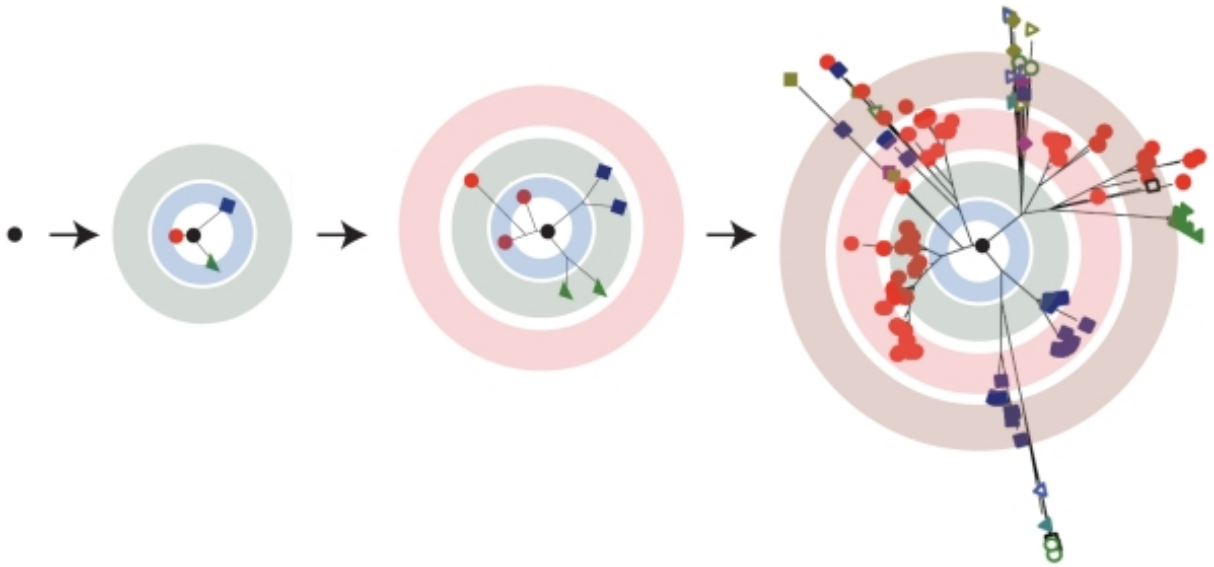


Figure 1.8 RNA viruses exist as a quasispecies. A single virion that infects a host will quickly branch out and create a diverse array of variants. It is thought that every possible point mutation, and many double/triple mutations, are generated at each viral replication cycle (Vignuzzi et al., 2005). This diversity is useful for viral adaptation, but can complicate computational analyses. Adapted from Lauring and Andino (2010).

homology-independent manner (Table 1.6). After this analysis, there may be many contigs identified to have a viral origin. Some of these may be from different viruses, while others are fragments of single viral genome. To create a draft viral genome, these fragments need to be joined or grouped.

Table 1.6 Computation approaches for detecting viral contigs.

Method	Description
Simple protein homology search by local alignment	Searches a single contig against a database of known sequences by local alignment. Common tools for this are BLASTx (Camacho et al., 2009), Diamond (Buchfink et al., 2015), MMseqs2 (Steinegger and Söding, 2017), RAPSearch2 (Zhao et al., 2012), and Kaiju (Menzel et al., 2016).
Position Specific Scoring Matrix (PSSM) search	Finds related sequences by local alignment to build a PSSM. This PSSM summarises significant features present in these sequences, which can be further used to search for more distant sequences. This is implemented in PSI-BLAST (Camacho et al., 2009) and MMseqs2 (Steinegger and Söding, 2017).

Table 1.6 Computation approaches for detecting viral contigs (cont.)

Method	Description
Profile Hidden Markov Model (HMM) search	Builds a profile HMM for an alignment of related sequences. This represents the sequences as probabilistic model, allowing a more sensitive homology search. HMMER (Eddy, 2011) can use a clustered group of unknown contigs to search a sequence database. vFam (Skewes-Cox et al., 2014) and ClassiPhage (Chibani et al., 2019) utilise pre-built HMMs from known viruses to classify contigs. HH-suite (Steinegger et al., 2019) allows an even more sensitive approach, by searching HMMs against each other.
Genome Relationships Applied to Virus Taxonomy (GRAViTy)	Aiewsakun and Simmonds (2018) have suggested the use of genomic organisation as a marker of viral families. This utilises databases of protein profile hidden Markov models (PPHMMs) and genomic organisation models (GOMs). PPHMMs are used to find viral genes in a contig, the locations of which are compared to the GOM of the match. This gives an increased search sensitivity while reducing false positives.
Taxonomic classification using k-mers	Kraken (Wood and Salzberg, 2014), MetaOthello (Liu et al., 2018), Clark (Ounit et al., 2015), and NBC (Rosen et al., 2008) break each contig down into k-mers. This is compared to a database of taxonomy-identifying k-mers to bin each sequence.
K-mer frequency bias	The frequency of 1- 2- or 3-mers in a contig give it a specific signature, which can be searched against other signatures to find related sequences (Trifonov and Rabadan, 2010).
Random Forest (RF) classification	RFs are a machine learning technique that utilise an ensemble of decision trees for classification or prediction. VirFinder (Ren et al., 2017) utilises this technique to classify viral contigs by their k-mers. MARVEL (Amgarten et al., 2018) improves on this by instead using a variety of features of predicted genes.
Convolutional Neural Network (CNN) classification	CNNs are a deep learning technique that was originally based on the structure of receptive fields in the visual cortex. DeepVirFinder (Ren et al., 2018) uses a CNN trained to identify patterns in a contig to classify whether it is of viral origin. ViraMiner (Tampuu et al., 2019) improves on this by using a branched model that integrates two different approaches.

Table 1.6 Computation approaches for detecting viral contigs (cont.)

Method	Description
Relative Synonymous Codon Usage frequency (RSCU)	Some amino acids are encoded by several synonymous codons, with the choice of codon for each of these amino acids, termed RSCU, differing among species. As viral RSCUs may differ from their hosts, (Bzhalava et al., 2018) trained an RF and a simple artificial neural network to discriminate between viral and non-viral RSCUs.
Novel protein family discovery	Barrientos-Somarribas et al. (2018) describe an approach that utilises the huge mutation rate of viruses for their identification, which produce many sequence variants during assembly. Contigs are clustered by similarity, with the largest clusters being searched for evidence of coding regions. Long, statistically significant coding regions that don't match known protein families are likely to be of viral origin.
Automated pipelines	A number of automated pipelines have been created that combine a variety of the approaches above. This includes VirusDetect (Zheng et al., 2017), ViraPipe (Maarala et al., 2018), VirusMeta (NIASC and NIASC, 2017), VIROME (Wommack et al., 2012; Zhao et al., 2012), VMGAP (Lorenzi et al., 2011), MetaVir (Roux et al., 2014), VirSorter (Roux et al., 2015), VirusSeeker (Zhao et al., 2017), VirusFinder (Wang et al., 2013), MetaSAMS (Zakrzewski et al., 2013), and VirFind (Ho and Tzanetakis, 2014).

1.2.9. Scaffolding and binning

Assembly merges reads together to create consensus contigs. Mis-assemblies, inconsistent coverage across the genome, and the presence of genome repeats prevent the creation of single contigs that span the whole of a genome (Mandric et al., 2018). Though viral genomes are short, their high variation and uneven read depth may still result in the generation of short contigs. Joining contigs together to form genomes or chromosomes is called scaffolding. One way to accomplish this is by using large-insert paired-end reads (Figure 1.9). If only single-end or small-insert paired-end sequencing is available, then the scaffolding process must be done using other information. The tools used for these are listed in Table 1.7.

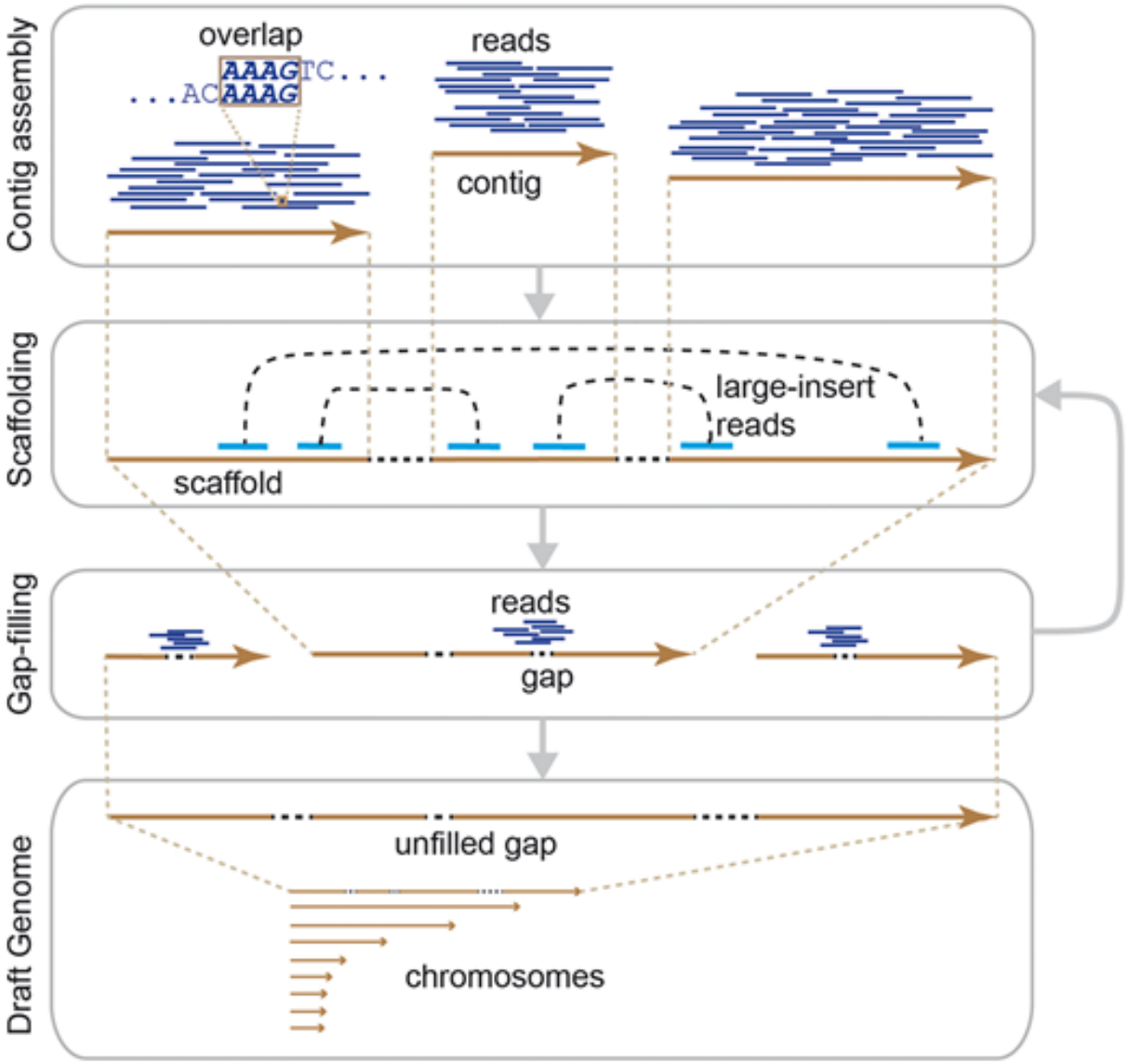


Figure 1.9 Workflow for the processing of de novo assembled contigs to a draft genome using paired-end reads. These reads are generated by having large fragments sequenced from both ends, one such method for is mate-pair sequencing. As second generation platforms generate short reads, this leaves a gap of roughly known length in-between the reads, known as the insert size. This information can be used to join contigs, as reads on different contigs are known to originate from the same molecule. Though joined, this generates a gap between contigs, which can be filled using previously discarded reads. Gaps may still be remaining, but this can often be enough to generate a draft genome. Taken from Sohn and Nam (2018).

Table 1.7 Scaffolding tools for the joining or binning of contigs.

Scaffolding tool	Utilised information
Reconciliator (Zimin et al., 2008)	Multiple assemblies
SOMA (Nagarajan et al., 2008)	Optical map
ABySS (Simpson et al., 2009)	Paired-end reads
SOAPdenovo (Li et al., 2010)	Paired-end reads
SOPRA (Dayarian et al., 2010)	Paired-end reads

Table 1.7 Scaffolding tools for the joining or binning of contigs (cont.)

Scaffolding tool	Utilised information
MAIA (Nijkamp et al., 2010)	Multiple assemblies and multiple reference genomes
Bambus 2 (Koren et al., 2011)	Contig graph
MIP (Salmela et al., 2011)	Paired-end reads
Opera (Gao et al., 2011)	Paired-end reads
SOAPdenovo2 (Luo et al., 2012)	Paired-end reads
GAA (Yao et al., 2012)	Multiple assemblies
GRASS (Gritsenko et al., 2012)	Paired-end reads
Mix (Soueidan et al., 2013)	Multiple assemblies
SCARPA (Donmez and Brudno, 2013)	Paired-end reads
SWiPS (Li and Copley, 2013)	Homologous proteins
GAM-NGS (Vicedomini et al., 2013)	Multiple assemblies
CISA (Lin and Liao, 2013)	Multiple assemblies
OMACC (Chen et al., 2013)	Optical map
iMetAMOS (Koren et al., 2014)	Multiple assemblies
SOAPdenovo-Trans (Xie et al., 2014)	Paired-end reads and read mapping
Ragout (Kolmogorov et al., 2014)	Multiple reference genomes
SSPACE-LongRead (Boetzer and Pirovano, 2014)	Long reads
Enly (Fondi et al., 2014)	Read mapping
FGAP (Piro et al., 2014)	Multiple assemblies
SILP2 (Lindsay et al., 2014)	Paired-end reads
BESST (Sahlin et al., 2014)	Paired-end reads
MeGAMerge (Scholz et al., 2014)	Multiple assemblies and read mapping
Slicembler (Mirebrahim et al., 2015)	Multiple assemblies
ScaffMatch (Mandric and Zelikovsky, 2015)	Paired-end reads
MeDuSa (Bosi et al., 2015)	Multiple reference genomes
NaS (Madoui et al., 2015)	Long reads

Table 1.7 Scaffolding tools for the joining or binning of contigs (cont.)

Scaffolding tool	Utilised information
Sealer (Paulino et al., 2015)	Assembly graph
GMCLoser (Kosugi et al., 2015)	Paired-end reads, long reads, or multiple assemblies
WiseScaffolder (Farrant et al., 2015)	Paired-end reads
Metassembler (Wences and Schatz, 2015)	Multiple assemblies
InteMAP (Lai et al., 2015)	Multiple assemblies
Multi-CAR (Chen et al., 2016)	Multiple reference genomes
MaGuS (Madoui et al., 2016)	Genetic map
PEP_scaffolder (Zhu et al., 2016)	Homologous proteins
GapBlaster (Sá et al., 2016)	Multiple assemblies
ScaffoldScaffolder (Bodily et al., 2016)	Paired-end reads
OPERA-LG (Gao et al., 2016)	Long reads
BIGMAC (Lam et al., 2016)	Long reads
ABySS 2.0 (Jackman et al., 2017)	Paired-end reads
BOSS (Luo et al., 2017)	Paired-end reads
RadMap (Dou et al., 2017)	Genetic map
Unicycler (Wick et al., 2017)	Long reads
CAMSA (Aganezov and Alekseyev, 2017)	Multiple assemblies
ARCS (Yeo et al., 2018)	Linked reads
McCortex (Turner et al., 2018)	Connectivity information in reads
Multi-CSAR (Chen et al., 2018a)	Multiple reference genomes
Tigmint (Jackman et al., 2018)	Linked reads
LR_Gapcloser (Xu et al., 2019)	Long reads

Contigs that cannot be joined may still be binned together, which is especially useful for viral genomes that travel as multiple nucleic acid strands. This can be done using a motif finding tool such as MEME (Bailey et al., 2015), that attempts to link contigs based on repeating sequences. Another way is to use their read coverage, codon usage, and/or k-mer spectra, such as with CONCOCT (Alneberg et al., 2014), GroopM (Imelfort et al., 2014), MetaBAT (Kang et al.,

2015), MyCC (Lin and Liao, 2016), CoMet (Herath et al., 2017), COCACOLA (Lu et al., 2017), IFCM (Lin and Liao, 2016; Liu et al., 2017), SolidBin (Wang et al., 2019), d2SBin (Wang et al., 2017), and BMC3C (Yu et al., 2018). By the end of this process, you should hopefully have a complete sequenced, assembled, scaffolded, and binned virome, ready for further analysis.

1.3. Conclusions

The study of plant viromes is a relatively new field, only having become established in the last decade thanks to the rise of high-throughput sequencing. The tools and techniques to handle this kind of data are varied, with many new approaches being developed each year, generating even more alternative approaches. This is most evident for computational tools, where the different types of information contained within reads, assembly graphs, and contigs can be exploited to produce distinct software that aims to produce the same output: an accurate characterisation of a whole virome.

1.4. Aims and objectives

This thesis focuses specifically on the software used for the detection of viral genomes within high-throughput sequencing datasets. This stage of viral metagenomics may be the most crucial - being unable to detect the presence of a virus/viroid within a dataset would be a detriment for viral surveillance and discovery. The diverse approaches taken by virus detection tools, especially the recently developed machine learning approaches, may give different tools advantages in certain scenarios - the detection of viral genomes at low read numbers, the detection of highly divergent viral genomes, or the detection of viruses which belong to taxonomic groups that are underrepresented within reference datasets. Importantly, these scenarios may all occur within a single virome. Finding the limits of each approach can inform us of their applicability, i.e. should you use certain tools based on the properties of your dataset? Should you use multiple approaches to cover all possibilities? Additionally, are there any scenarios where we are completely unable to detect the presence of a viral genome in a dataset using current approaches?

To attempt to answer these questions, we aim to benchmark current software in viral detection to find the factors influencing their limits of detection. In order to achieve this, we must first develop accurate metrics by which to define these limits. While this is simple for virus/viroid read numbers, as we can choose the number of reads we include in a test, and relatively straightforward to control the sparsity of taxonomy, there is no simple technique that can give an accurate estimate of how distant two highly divergent viral genomes are. We therefore aim to develop a tool that can calculate this metric. We then apply this metric to find the limits of software used in the detection of plant viruses/viroids in high-throughput sequencing datasets. Finally, we use this knowledge to analyse real sequencing datasets, to uncover whether there is a difference in what these tools consider viral, and to find whether there is utility in a combined approach.

1.5. Thesis outline

This thesis is divided into five additional chapters. Chapter 2 describes the materials and methods used in subsequent chapters. Chapter 3 presents and tests our bespoke software for accurately defining the distance between highly divergent viral genomes. Chapter 4 investigates the factors influencing the limits of detection of current viral detection approaches. Chapter 5 involves the application and analysis of benchmarked tools to previously seen and novel sequencing datasets. Finally, Chapter 6 summarises the findings of this study, discusses their significance, and presents an outlook for future work.

Chapter 2. Materials and methods

This Chapter details the computational environment, tools and methodologies shared across all data chapters. Specific methodologies related to results are detailed in the relevant chapters.

2.1. Platform specification

All programs were developed and executed on the Newcastle University Rocket HPC platform (Newcastle University IT Service, 2023). When benchmarking programs, 'Standard' compute nodes are used, with the following specifications:

- 2 Intel Xeon E5-2699 v4 processors (2.2 GHz, 22 cores, 55 MB cache)
- 44 cores (2 processors * 22 cores)
- 128 GB memory - (8 DDR4 RDIMMs, each with 16GB)
- 600 GB SAS disk (469 GB scratch space)
- CentOS 7.9

Each program is run either sequentially on a single node, or separately on different nodes, utilising all cores when possible.

2.2. Availability of code and environments

Full specification of the Conda environment (Anaconda, 2016) used for development and testing is listed in Appendix A. Python 3.9.16 (Van Rossum and Drake, 2009) code for *Mottle* and *Mottle-map* is shown in Appendix B, and is made available online at https://github.com/tphoward/Mottle_Repo.

2.3. Generation of Rfam concatenated aritficial genomes

Artificial viral genomes were constructed by the concatenation of short sequences taken from the seed alignments used for generating Rfam models (Kalvari et al., 2021). Firstly, Rfam seed sequences, as well as family phylogenetic trees in Newick format (Felsenstein, 2022), are obtained from the Rfam FTP site (Rfam team, 2023) (fetched 2021-07-05). Within each seed alignment, taxonomic identifiers (taxID) are taken for each sequence from the tree file, then their full taxonomic lineages were obtained using the ETE3 toolkit (Huerta-Cepas et al., 2016), and

only those of viral origin (taxID of 10239) are kept for further processing. From each family, now only containing virus or viroid sequences, pairwise alignments are extracted, and substitution distances are calculated from Newick tree branch lengths using ETE3. Generation of artificial genomes is then done with the following protocol:

- A target substitution distance (substitutions per base pair) is chosen.
- Find the sequence pair with the closest substitution distance to the target. Set these as the initial genome sequences and remove them from the search pool. If there are multiple pairs that are equally close to the target, choose one pair at random.
- If the mean substitution distance across the artificial genomes is above the target, find the sequence pair with the closest distance to the target that is below the target, or else find those closest that are above. If the mean distance is exactly equal to the target distance, simply choose the closest. Join these onto the end of the artificial genomes.
- Repeat the above step until the length of alignment of the artificial genomes reaches 4,000 base pairs, or the pool of sequence pair above or below the target distance is exhausted.

2.4. Viral read detection protocols

Six programs were used for viral read detection in Chapters 4 and 5: MMseqs2 (Mirdita et al., 2021), HMMER3 (Wheeler and Eddy, 2013), DeepVirFinder (Ren et al., 2020), Mash Screen (Ondov et al., 2019), GraphAligner (Rautiainen and Marschall, 2020), and PathRacer (Shlemov and Korobeynikov, 2019). Each of these tools operate with different expected inputs, and produce different outputs. In order to standardise their processes to allow comparison, we use a read-centred approach. Read sets are processed to a format that can be used as tool inputs, and program output scores are propagated back to the read set.

For contig-based software (MMseqs2, HMMER3, and DeepVirFinder), reads are first processed by Fastp (Chen et al., 2018b) to remove adapters and for quality trimming. They are then assembled into contigs by RnaviralSPAdes (Meleshko et al., 2021). The reads are mapped back to the contigs by Bowtie2 (Langmead et al., 2019). Then the tools are executed, with these contigs as input. From their outputs, bit-scores are extracted, as these are independent of reference database size. Contigs that had no hits, or had their highest-scoring hit to non-viral genomes have their scores set to zero, otherwise their score is set to the highest bit-score. Finally, contig scores are propagated back to the mapped read set. MMseqs2 and HMMER3 are both executed using nucleotide and amino-acid search, with the former using nucleotide and translated nucleotide sequences, and the latter using nucleotide (Rfam) (Kalvari et al., 2021) and amino-acid models (Pfam) (Mistry et al., 2021) models. The highest scoring hit for each read, between the two search types, before zero-ing is set as the primary hit. DeepVirFinder can only read nucleotide input, and so no amino-acid conversion is used.

For graph-based software (GraphAligner and PathRacer), reads are processed by Fastp to remove adapters only. de Bruijn graphs are built from the reads using spades-gbuilder (Center

for Algorithmic Biotechnology, 2022). The tools are executed on these graphs, returning either corrected reference sequences (GraphAligner), or the graph sequences that mapped to the reference models (PathRacer). Reads are then mapped to these outputs, returning mapping quality (MAPQ) values for each read. Unmapped reads, and those that map to non-viral sequences have their MAPQ set to zero. PathRacer is able to use the same Rfam and Pfam models as HMMER3, and so both nucleotide and amino acid search is performed. GraphAligner can only be executed on nucleotide graphs, so no amino-acid search is performed.

Mash Screen is executed directly on reads, using k-mer databases constructed from reference genomes. Two sets of reference k-mer databases are used, one that contains only non-viral genomes, and one that contains only viral genomes. These are created for both nucleotide and translated nucleotide sequences, giving a total of four databases. Reads are first processed, by Fastp, to remove adapters only. Mash Screen is then executed on these reads for each database, returning p-values for containment within the viral and non-viral databases. P-values are converted to bit-scores with the following formula:

$$bitscore = \log_2 pvalue$$

Reads were assigned the highest scoring hits, and those with non-viral hits had their score reverted to zero.

Parameters used during the execution of each command are as listed follows:
Fastp: `--detect_adapter_for_pe`, `--include_unmerged`, `-GALQ` when trimming is used, `--thread 44`
SeqKit translate: `-f 6`, `--clean`, `-j 44`
RnaviralSPAdes: `-t 44`
Spades-gbuilder: `-k 15`, `-t 44`
Mash sketch: `-i`, `-k 15` for nucleotide and `-a -k 7` for amino acid search, `-p 44`
Bowtie2: `--very-sensitive-local`, `--omit-sec-seq`, `--threads 44`
MMseqs2: `easy-search`, `-s 7.5`, `-e 10`, `--search-type 3` for nucleotide `--search-type 2` for amino acid search, `--threads 44`
HMMER3: `-E 10`, `--cpu 44`
DeepVirFinder: `-l 44`, `-c 44`
Mash screen: `-p 44`
GraphAligner: `--preset dbg`, `--seeds-minimizer-length 15`, `--seeds-minimizer-windowsize 30`, `--E-cutoff 10`, `--precise-clipping 0.501`, `-t 44`
PathRacer: `--parallel-components`, `--rescore`, `-t 44`

Chapter 3. Creating an Accurate Pairwise Distance Estimation of Highly Divergent Viral Genomes

Summary: Current tools for estimating the substitution distance between two related sequences struggle to remain accurate at a high divergence. Difficulties at distant homologies, such as false seeding and over-alignment, create a high barrier for the development of a stable estimator. This is especially true for viral genomes, which carry a high rate of mutation, small size, and sparse taxonomy. Developing an accurate substitution distance measure would help to elucidate the relationship between highly divergent sequences, interrogate their evolutionary history, and better facilitate the discovery of new viral genomes. To tackle these problems, we propose an approach that uses short-read mappers to create whole-genome maps, and gradient descent to isolate the homologous fraction and calculate the final distance value. We implement this approach as Mottle. With the use of simulated and biological sequences, Mottle was able to remain stable to 0.66 - 0.96 substitutions per base pair and identify viral out-group genomes with 95% accuracy at the family-order level. Our results indicate that Mottle performs as well as existing programs in identifying taxonomic relationships, with more accurate numerical estimation of genomic distance over greater divergences. By contrast, a limitation is a comparatively lower numerical accuracy at low divergences, and on genomes where insertions and deletions are uncommon. We propose that Mottle may therefore be of particular interest in the study of viruses, viral relationships, and notably for viral discovery platforms, helping in benchmarking of homology search tools and defining the limits of taxonomic classification methods.

3.1. Introduction

Pairwise nucleotide substitution distance is widely used in bioinformatic analyses. Pairwise comparisons within collections of genomes are commonly integrated to establish phylogenies, providing insight into their shared evolutionary history. They are similarly used to position novel genomes within an established phylogeny (Zaharias and Warnow, 2022). Substitution distances can also be converted to genome-wide percentage identity to define taxonomic demarcations, such as the distance a novel genome must be from known genomes to establish it as a new species (Breitbart et al., 2017). Comparing the distances of discrete genetic particles such as chromosomes, plasmids, plastids, and segments can find differences in their evolutionary histories, elucidating how genetic material has been exchanged between organisms or populations. Finding the most distinct or representative sequences in a set is used to create a compressed database for homology search and taxonomic classification (Tang et al., 2019). A

high substitution distance can however be a large barrier for sequence discovery (Krishnamurthy and Wang, 2017). Being able to accurately define this distance allows improved benchmarking of homology search and taxonomic classification tools in order to find their limits, allowing accurate estimates of what may pass through the silicon sieve of *de novo* sequencing-based discovery and diagnostics.

Despite the considerable number of approaches available for pairwise genome comparison (Zielezinski et al., 2019), many do not produce a biologically-relevant substitution distance, and from those that do, there are few that are suitable for highly divergent sequences, i.e., those that have had many substitutions per site between them. This is for two main reasons – work has focused on creating faster and more efficient tools that work well at low divergences (Baker and Langmead, 2019; Girgis et al., 2021; Klötzl and Haubold, 2020; Leimeister et al., 2019b; Uddin et al., 2022; Zhao, 2019), combined with the inherent difficulties found when making comparisons at high divergences. The largest of these difficulties are false seeding and over-alignment. Seeding is the process of finding small subsequences, often in the tens of bases, that are identical or near-identical in two sequences and can be extended to larger regions of homology. Finding seeds at non-homologous locations, i.e., false seeding, may add regions that otherwise have little similarity into the distance calculation, artificially inflating it. At low divergences a large seed size may be used as there would be few mutations between homologous regions. At increasing divergence a smaller seed size is needed to find such regions, generating many seeding locations that are spurious, eventually overwhelming the limited number of truly homologous seed sites. Finding seed locations that have true homology while avoiding or removing false ones becomes a critical task at distant homologies. While false seeding can make sequences appear more divergent than they truly are, over-alignment can make them appear more similar. Alignment is usually done after seeding, inserting gaps into either sequence to match up homologous nucleotides that were shifted due to insertions and deletions (InDels). This presents the danger of over-correcting for InDels, falsely pairing matching nucleotides that are actually non-homologous, such as mistaking adjacent substitution events as an insertion, and therefore not contributing to the substitution distance. Avoiding over-alignment while still correctly aligning regions can be difficult at a low divergence, but at a high divergence, where multiple different mutation events may have occurred at the same nucleotide position, it may not be possible.

These difficulties appear especially often when studying viral genomes, due to their high rate of mutation, small size, lack of universal marker genes, and the low proportion of known viruses (Duffy et al., 2008; Santiago-Rodriguez and Hollister, 2022). Viral replication machinery, especially those of single-stranded RNA based viruses, are known to introduce many substitutions every generation. Combined with a short generation time, these viruses can quickly diverge from their progenitor genome. Additionally, the ratio of InDel event to substitution events is extremely high in these genomes (Sanjuán et al., 2010), which makes finding true seed more difficult and over-alignment more likely. The usually small size of viral genomes further reduces the number of possible seeding sites. Finally, the number of viral species that have been

documented is a small proportion of the total estimated number of viral species (Koonin et al., 2023), with estimates of the proportion of characterised orthornaviran RNA viruses being estimated as low as 0.006% (Dominguez-Huerta et al., 2023), and environmental sampling projects discovering many previously unobserved viral genomes with each sequencing experiment (Neri et al., 2022). The effect of sparse taxonomic coverage is that many viral genomes are highly diverged from any known viruses, making it difficult to study their relationship to other viruses. To this end, viral genome analysis is the field that may benefit the most from improvements in pairwise sequence distance accuracy at high divergences.

In the face of these difficulties, current tools attempt to alleviate parts of the false seeding or over-alignment problems: some programs attempt to find non-exact matches for seeding, finding longer seeds which are more likely to be true seeds (Leimeister et al., 2019a). Others have sophisticated algorithms for distinguishing between true and false seeds (Hachiya et al., 2009). Programs that do not use any alignment step, known as alignment-free programs, are able to completely avoid the problem of over-alignment, instead using statistical information from across a sequence that is correlated to substitution distance – e.g., proportion of shared k-mers (Criscuolo, 2020), shortest unique substrings (Haubold et al., 2009), average common substring (Leimeister and Morgenstern, 2014), or sequence embeddings (Zheng et al., 2019). This creates a separate issue – as sequence divergences increase, the correlation between these global statistical data and the local nucleotide substitutions they estimate can become increasingly decoupled. A new approach is needed, one which creates high-quality seeds and avoids over-alignment while incorporating direct nucleotide-level information.

Short-read mappers (henceforth referred to as mappers) are tools used to map short sequence fragments of 50-300bp, produced from high-throughput sequencing runs, to reference genomes. These mappers already employ sophisticated seeding algorithms, give many parameters to tune mapping sensitivity, can use arbitrary queries and references, and are designed to handle thousands to millions of fragments. Sequence fragments created *in silico* can therefore be used as inputs to these mappers, finding the optimal homologous location for each fragment on another sequence, and allowing downstream processing to calculate substitution distance. This approach of running mappers on in silico fragments has been successfully utilised in other bioinformatic applications – for generating multiple-sequence alignments in ViralMSA (Moshiri, 2021), and for constructing phylogenies in REALPHY (Bertels et al., 2014). In this chapter we explore the application of these mappers for estimating pairwise nucleotide substitution distance, how careful use of their outputs can avoid false seeding and over-alignment, and describe the implementation of Mottle - a tool for more accurate distance calculation between two highly divergent sequences.

3.2. Design and Implementation

Mottle takes two arbitrary nucleotide sequences of unknown relation, and outputs an estimated substitution distance between them. This can make use of any mapping software that aligns

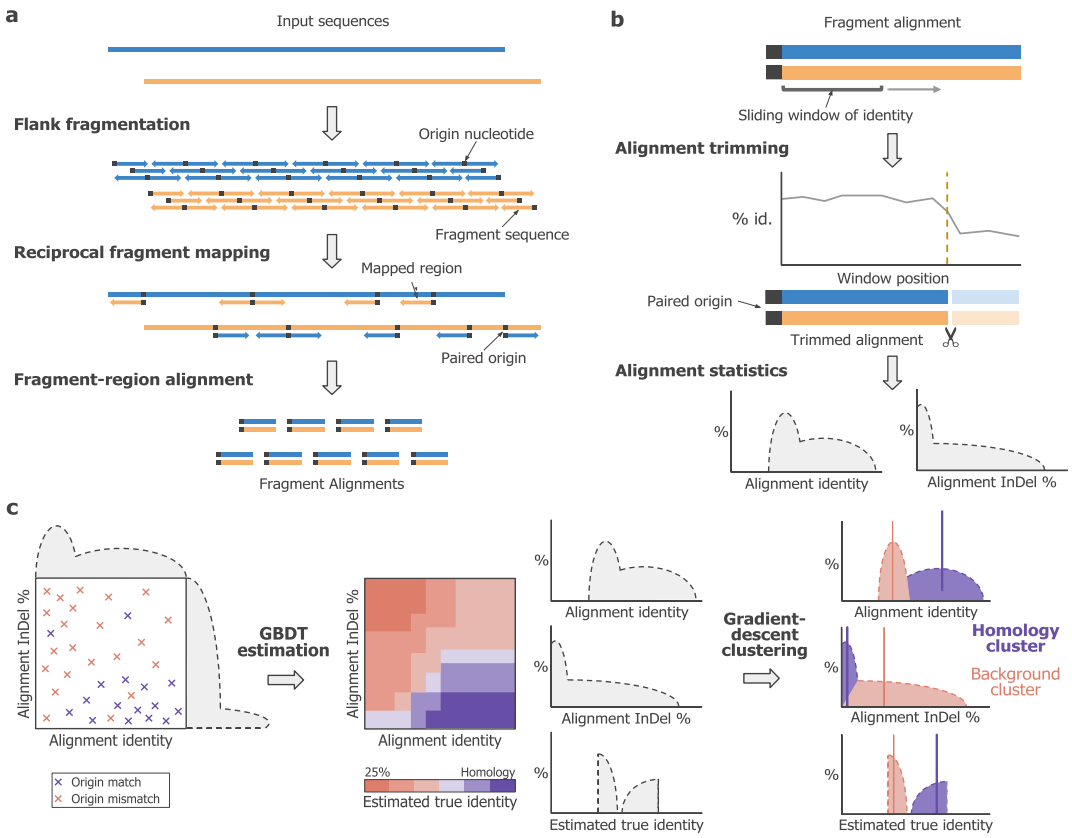


Figure 3.1 Overview of Mottle's sequence distance estimation algorithm. **(a)** Generating fragment alignments from input sequences. Each sequence is fragmented *in silico*. The origin nucleotide is excluded from each fragment sequence. Fragments are mapped onto the reciprocal sequence via a mapper, with each mapped fragment's origin being paired. Origin pairs carry a binary state (match or mismatch). Fragment sequences are then fully aligned. **(b)** Trimming alignments on identity change. For each alignment, a sliding window calculates percentage identity. If a window's identity diverges from the initial window's, all nucleotides from that point onwards are discarded. **(c)** Fragment clustering and substitution distance estimation. For each alignment, identity and InDel percentage statistics are calculated. These are fed into a Gradient Boosted Decision Tree (GBDT), which is trained to predict origin pair match state. This gives a predicted match probability on each alignment that can be interpreted as a bias-free identity. These three statistics are used for gradient-descent clustering, to find a cluster of alignments that were generated due to shared homology, and a cluster for those due to chance. Once both fractions are obtained, a mean origin identity is calculated for the homology cluster, which is used to derive the final substitution distance between the two sequences.

short fragments to larger sequences. Additionally, we have implemented a bespoke fragment mapping algorithm for this process, *Mottle-map*, which guarantees that each fragment is mapped but is not scalable to large sequences. Both algorithms are described in the subsections below.

3.2.1. Mottle: Calculating pairwise sequence distance from mapped fragments.

The main Mottle program can be split into three stages – fragment generation, alignment processing, and alignment clustering, presented in Fig 3.1 a to c respectively and described in the following sections. Briefly, for each position, p, in query genome, q, we set the nucleotide at that position, q_p, as the origin nucleotide. The set of flanking regions either side, $q[p+1..p+n]$ and $q[p-1..p-n]$ for a specified flank size n, of each origin nucleotide are mapped using a short-read mapper to a similar set of flanking regions in the target genome. For each mapping,

we calculate flank alignments, and gather a set of statistics: the alignment identity, the InDel rate of the alignment, and a binary value representing whether the corresponding origin nucleotides match or mismatch. We represent mappings as points defined by identity and InDel statistics, labelled by match state, from which we can calculate a match probability that acts as an estimator for true identity without alignment bias. These points are clustered into two sets to separate homologous mappings from spurious ones, and genomic distance is estimated from the homologous set.

Similarities and differences to standard approaches

There is a wide breadth of approaches that have been utilised for pairwise substitution distance estimation, which Mottle builds upon, but also introduces novel differences. Firstly, Mottle generates a distance estimate based on the similarity or dissimilarity between pairs of nucleotides rather than a proxy statistics, so it falls within the alignment or micro-alignment based category of programs rather than alignment-free approaches. The major theoretical differences between Mottle and other such approaches are found in the flank fragmentation and gradient-descent clustering stages.

A naive alignment based approach, such as averaging BLAST (Camacho et al., 2009) alignment identities, would suffer greatly from over-alignment, as the bases that are used to calculate distance are the same bases used to generate the alignment. Micro-alignment approaches, such as Co-Phylog (Yi and Jin, 2013), remedy this by separating the 'context' used to find the micro-alignments from the 'objects' used to calculate the distance metric. The downside of micro-alignments, though, is a limited number of insertions or deletions, if any, within their alignments. Mottle uses a separation between the 'origin-nucleotide', equivalent to the 'object', and the 'flank-sequence', equivalent to the 'context', while allowing that calculation of full alignments, and so finding more 'objects' to compute substitution distances from at the greater InDel rates seen in viral genomes.

Standard alignment based approaches may use simple cut-offs or genomic statistics to decide which 'objects' to use for distance calculation and which to discard, or may simply use all found alignments and attempt to correct for the inclusion of non-homologous 'objects'. Mottle introduces the use of gradient-descent for deciding which alignments to keep and which to discard. Rather than finding a simple score cut-off, below which alignments are discarded, mottle uses multiple alignment statistics to divide alignments into a 'homologous' cluster and 'non-homologous' cluster. This can theoretically allow highly divergent sequences, which would normally have few, if any, high-scoring alignments, to still produce a valid estimate. Additionally, this may reduce the bias introduced by selecting for high-scoring alignments, which are more likely to have smaller substitution distances.

Fragment generation

Flank fragmentation: Each sequence can be separated into a discrete set of subsequences, each defined by a unique combination of origin nucleotide, subsequence size, and relative direction. Each nucleotide in the full-length sequence can act as an origin - the first nucleotide of a subsequence - of multiple subsequences. But the possible sizes and relative directions of these subsequences are restricted depending on the origin nucleotide's location. If subsequence size is kept fixed, each nucleotide is an origin for up to two subsequences, for each of the forward and reverse complement directions. The portion of each subsequence that does not contain the origin nucleotide is termed a fragment, and the two possible fragments for each origin are termed the forward and reverse flank fragments of the origin. Considering the origin nucleotide separately from the fragments is integral in estimating true identity in the later stages of the algorithm.

Reciprocal fragment mapping: This step takes the two full-length input sequences, their generated fragments, and a mapper program, to create a mapping between each fragment and a homologous area in the sequence of comparison. To allow the use of a variety of mapping algorithms, there are multiple modes of input depending on what the mapper accepts – sequence and fragments, fragments and fragments, or sequence and sequence. The mappings are done reciprocally – one sequence is used as the query with the other acting as the reference, and vice versa. The output of a mapper is a series of possibly homologous regions between the two inputs. This must either contain fragment names or sequence locations to allow identification of origin nucleotide, query location, and mapped location for each region. In cases where a mapper returns multiple regions for a query, all mapping is kept. The mapped region sequence pairs and corresponding origin nucleotides are extracted for use in later steps.

Mapped region alignment: While some mappers output a full alignment between mapped regions, many do not. To get consistent and well-defined alignments, Needleman-Wunsch global alignment is carried out between region sequences independent of mapper. Aligned regions that contain gaps in the first N bases are discarded, where N is a parameter of Mottle, as these gaps may frame-shift the start of the alignment and therefore the origin nucleotide.

Alignment processing

Alignment trimming: Not all alignments will contain a consistent homology throughout. Some may begin with a high degree of similarity, but with a genomic rearrangement or large InDel creating a discontinuity that suddenly reduces similarity. This non-homologous section would change alignment properties, adding noise to statistics calculations, and confounding downstream clustering. To remove these discontinuities, a sliding window is moved through the alignment. The length and identity, the proportion of aligned nucleotides that match, of the first window is used to estimate a binomial distribution for match/mismatch states. Where a later window's identity is above or below that which would be expected by chance of this distribution, the alignment is clipped, discarding proceeding nucleotides. Clipped alignment shorter than a

minimum size are additionally discarded. The window size, two-tailed binomial test p-value, and minimum clipped alignment size are configurable parameters.

Alignment statistics calculation: To distinguish homologous from non-homologous alignments, a set of statistics is calculated for each of the non-discarded windows produced during trimming. Alignment identity is the fraction of matches in non-gap positions, corrected for the GC composition of the alignment. The fraction of gap positions in the window could be used as another such statistic, where homologous alignments would contain fewer gaps at the same identity. This would, though, not inform us of the probability that the origin nucleotide is shifted by an InDel event, as each event can insert or delete multiple nucleotides in a row.The number of InDel events in a row between non-gap positions can be approximated by a Geometric Distribution with PMF,

$$P(L = l) = p^l \cdot (1 - p)$$

where l is the number of adjacent InDel events between two non-gap positions regardless of the size of each event, $P(L)$ is the probability of finding this number of events between two arbitrary sites in the sequence, and p is the fraction of InDel events per site. p is directly related to the probability of no InDel events having occurred between two adjacent homologous nucleotides via $P(L = 0) = 1 - p$, which can be estimated from the fraction of adjacent aligned nucleotides that do not have gaps, q, in an observed sequence, $p = 1 - q$. This produces a set of identities and InDel fractions for each alignment - one for every fixed-size window within it. After this step, the alignments themselves are discarded, with only these statistics and the origin nucleotide match state being kept for the final steps.

Alignment clustering

GBDT estimation of true identity: Identity and InDel fraction can greatly vary between and within homologous alignments. For clustering to be effective, a more stable statistic needs to be calculated. For this purpose, we train a Gradient-Boosted Decision Tree (GBDT) to estimate a 'true identity' for each window over the space defined by the previous statistics (Fig. 3.1c, 'GBDT estimation'). This treats each window as a single point of data, with parameters identity and InDel fraction, and with prediction value being origin nucleotide match state (1=match, 0=mismatch). The GBDT creates a stepwise manifold over this space, which we enforce to be monotonically increasing with window identity and decreasing with InDel fraction. This estimates the proportion of origin nucleotides that match within each area of parameter combinations, which we term the true identity estimate. As a wide area can have the same value estimated, similar alignments are likely to have windows that share similar true identities, making clustering more stable.

Gradient-descent clustering: The final step in Mottle is to isolate the homologous fraction of alignments through clustering. This approach involves the estimation of two cluster centres, one for homologous alignments (homologous cluster) and one for non-homologous alignments (null

cluster), within the three-dimensional space defined by Alignment Identity, InDel Fraction, and True Identity statistics. For every alignment, the mean and variance of each window statistic is calculated. This defines a normalised distance to each cluster centre as,

$$dist = \sqrt{(\frac{mean_I}{var_I})^2 + (\frac{mean_D}{var_D})^2 + (\frac{mean_T}{var_T})^2}$$

With $mean_I$ and var_I equal to the mean and variance of alignment identity, $mean_D$ and var_D of InDel fraction, and $mean_T$ and var_T of estimated true identity. The centre of the null cluster is initialised so that its True Identity is equal to the expected proportion of matches if it was due to chance, i.e. 0.25 if the GC-content is 50%, with the Alignment Identity and InDel fraction being set to the values of the alignment with the closest mean True Identity. The centre of the Homologous cluster is initialised to the values of the alignment at the 90th percentile of mean True Identities. Once Initialised, the cluster centres are shifted via gradient descent to reduce normalised distances between alignments and centres, via the loss function,

$$\text{loss} = \frac{\sum \text{dist}^2 \cdot \text{weight}}{\sum \text{weight} - 1} \cdot \text{learn\_mult}$$

where

$$weight = \left| \frac{\text{simils} \times 2 - 1}{2} + 0.5 \right|^{\left(\frac{1}{\text{binpow}}\right)} \times \text{sign}\left(\frac{\text{simils} \times 2 - 1}{2} + 0.5 \right)$$

and

$$simils = \frac{1}{\max(\text{dists}, \text{eps})}$$

which gives the highest weights to the nearest centres. Learn_mult, binpow, and eps are program parameters. A BFGS optimiser is used to find the locations that give the minimum loss values. Alignments are then each assigned to the cluster with the closest centre. The final distance value, D, returned by Mottle is obtained from the origin match proportion of alignments in the homology cluster, corrected for GC-content, and transformed to an estimate of substitution distance via the Jukes-Cantor model (Jukes and Cantor, 1969),

$$D = -\frac{3}{4} ln(1 - \frac{4}{3}(1 - I))$$

with I being equal to mean true identity of the homologous cluster.

3.2.2. Bespoke fragment mapping algorithm

While Mottle may use any short-read mapper for fragment mapping, a bespoke algorithm was developed to ensure that each position in a sequence is mapped even at large divergences. Mottle-map achieves this by transforming each fragment to a high-dimensional embedding via the Fast Fourier Transform (FFT) and subsequently finding the nearest neighbour in the reciprocal sequence. This is similar to the approach Satsuma takes for synteny detection (Grabherr et al., 2010). To allow the FFT transformation of fragments, their values must first be mapped to the complex number plane. For this we use the same embeddings as MAFFT (Katoh

et al., 2019), with each base placed at axis-aligned unit lengths and bonding pairs placed in opposite sides of the origin $G \to +1, C \to -1, A \to +i, T/U \to -i$. If each position in the fragment is treated as an embedding dimension, then two sequences that are more similar will have a smaller Euclidean distance between embedding, as long as there are no InDels, which would give a mid-sequence shift that would misalign the embedding. Shifting this embedding into frequency space via the FFT allows a Euclidean distance calculation while allowing for InDels. Before transformation, we divide the real and imaginary axes by the mean of their absolute values, to correct for GC-content. Afterwards, the frequency embeddings are normalised to unit L_2-norm. As Euclidean distance between frequency embeddings can be used as a dissimilarity metric between fragments, a nearest-neighbour search can be used between two sets of embeddings to find similar fragments. Since Mottle-map is made for small and highly-divergent sequences, we use a simple many-to-many comparison where the distance between each embedding vector is computed. The top N nearest-neighbour mappings for each fragment in two fragment sets are returned by Mottle-map, where N is a parameter.

3.2.3. Implementation details

Mottle and Mottle-map are implemented in Python 3.9.16, the environment used for its development and testing is described in Chapter 2, with all packages and versions listed in Appendix A. Full code is printed in Appendix B, as well as being available online at https://github.com/tphoward/Mottle_Repo. Parameters used in testing were *ntrees*=100, *nleaves*=32, *learn_rate*=0.1, *subsamp*=1.0, *binpow*=64, *lean_mult*=0.001, *reltol*=1e-20, *maxiter*=100, *binthres*=0.75, *prior_size=10*.

3.3. Materials and methods

In order to evaluate the performance of Mottle, we run a set of benchmarks where exact or approximate relationships between sequences are known. We run Mottle with either Mottle-map or Bwa-Mem2 as the mapper. As a comparison, we include four other programs that are used for substitution distance calculation - Co-phylog (Yi and Jin, 2013) which utilises micro-alignments between sequences, Mash which calculates distances based on shared k-mers (Ondov et al., 2016), Slope-SpaM that uses spaced-word matches (Röhling et al., 2020), and Swipe that calculates Smith-Waterman local alignments between sequences (Rognes, 2011). For the programs that output identities or alignments, a Jukes-Cantor (JC) model (Jukes and Cantor, 1969) is used to convert to substitution distances.

3.3.1. Simple sequence evolution

The first benchmark we use is a simple substitution model. We take the genomic sequence of Tobacco mosaic virus (RefSeq NC_001367.1), and introduce a set number, N_{sub}, of random substitution events *in silico*, following $N_{sub} = N_{nuc} \cdot D_{sub}$, with N_{nuc} being equal to the size of the full sequence and D_{sub} being the desired substitution distance. Multiple substitution events are

allowed to occur at each site. We then run each substitution distance program on the original and modified genomes to find a calculated distance. This is run for every 0.02 substitution per base-pair (sub/bp) distance between 0 and 1 inclusive. To have a measure of when the outputs of a program begin to consistently diverge from the true distance, that is independent of the number of trials, we calculate the Mean Cumulative Deviation (MCD) from the true distance,

$$dev = \frac{\sum_{k=0}^{N} |T_k - P_k|}{N}$$

where dev is the calculated MCD, T_k being equal to the target substitution distance of a specified trial, P_k the output of a substitution distance program within this trial, and N being equal to the total number of trials. As a threshold, we use an MCD value of 0.05 sub/bp, where the furthest distance a program's output is within the MCD threshold is its maximum stable distance.

3.3.2. Known family alignments

While a simple substitution model can give an upper bound on the maximum stable distance for each program, InDels are a common feature in biological sequences, especially those of viruses. To test how well Mottle can handle InDels, we endeavoured to create a benchmark that incorporates *in vivo* substitutions and InDels, while allowing us to know the ground truth in terms of substitution distance, and giving a large range of such distances. A database that allows us to do this is the RNA families (Rfam) database (Kalvari et al., 2021). Rfam catalogues homologous RNA sequences as families, and holds multiple sequence alignments of them. The process of generating Rfam concatenated genomes is described in Chapter 2, and the resulting sequence pairs are printed in Appendix C. We run this benchmark for the same distances as Section 3.3.1, and calculate MCDs in the same process.

3.3.3. Known genome taxonomies

Our final benchmark is to run all programs on full-length viral genomes, comparing the returned substitution distance with known taxonomies. While a true substitution distance between all viral genomes is not known, taxonomic relationships between genomes are generally based on their genetic similarities (Lefkowitz et al., 2018), where two genomes that share a lower rank are likely to be closer in substitution distance than those that only share a higher rank e.g. genomes in the same genus are likely to be closer than in the same family but not genus. To carry out this benchmark, we first select a test rank. We then randomly select a reference viral genome from the NCBI taxonomy database (Federhen, 2012) that shares this rank (but not below) with at least one other genomes, and the same for the rank above. From these sets, we randomly select a comparator genome and an out-group genome, respectively. On these, we run each program twice - once to calculate reference/comparator distance and once for reference/out-group. If the distance to the out-group is larger than to the comparator, this is recorded as a correct output, if smaller then incorrect, and if the output is the same for both then it is recorded as ambiguous. Some programs return NaN values or error codes if they are unable to find any homologous sites

to calculate a distance from. In this case, the values are recorded as the maximal distance, which we store as infinity, and carry out the comparisons as before. The rank comparisons we used were Genus-Family, Family-Order, Order-Class.

3.4. Results

Overall benchmarking results are summarised in Table 3.1.

Tool name	SUB	RFAM	GEN-FAM	FAM-ORD	ORD-CLASS
Slope-spam	0.52	0.22	0.92	0.70	0.37
Co-Phylog	0.28	0.12	0.82	0.38	0.12
Mash	0.44	0.18	**0.98**	0.85	**0.79**
Swipe	0.36	0.28	0.89	0.69	0.33
Bwa-Mem2	0.92	0.24	0.94	0.71	0.61
Mottle	**0.96**	**0.66**	0.95	**0.95**	0.75

Sub - Maximum stable distance of each program when tested on a simple *in silico* sequence evolution benchmark (section 3.3.1). Rfam - Maximum stable distance on concatenated RNA family alignments (section 3.3.2). Gen-fam, Fam-ord and Ord-class - Proportion of correctly assigned out-group genomes when comparing genomes in the same Genus/Family, Family/Order and Order/Class respectively (section 3.3.3). Scores in bold represent the best performing program in a benchmark.

3.4.1. Simple sequence evolution

Our first goal was to test how well Mottle performed in comparison to other programs designed to calculate pairwise substitution distances, in the absence of confounding factors. Specifically, we tested programs against the Tobacco mosaic virus genome, with a number of substitutions introduced *in silico* to a set substitution distance, and monitored their effectiveness at estimating this distance over increasing divergence. The results indicated that many of the tools began to struggle to calculate the true distance between 0.25 and 0.35 substitutions per nucleotide base (Fig. 3.2A, Table 1). An erratic behaviour was seen in many as they passed a critical threshold, set at 5% mean cumulative deviation from the true identity (Fig. 3.2B). Swipe and Mash however tended to significantly underestimate true distance past this point. Interestingly, the results for mash were incredibly stable after much deviation from the true distance (Fig 3.2A). The results indicated that Mottle, implemented with either Mottle-map or Bwa-Mem2 as the mapper, was able to accurately calculate the true distance over nearly all the sequence divergences tested, only reaching the 5% threshold towards the upper end of the testing space (Fig. 3.2B, Table S2). Noticeable but non-critical deviations were observed in Bwa-Mem2 and Mottle-map throughout the testing space (Fig 3.2A, Fig 3.2B), such that a stricter threshold would have been surpassed at a lower divergence. The results for existing tools were surprising, and it was expected that they would perform reliably over a greater distance. It should be noted, however, that the divergence tested here (up to 1 substitution per base) would be considered high

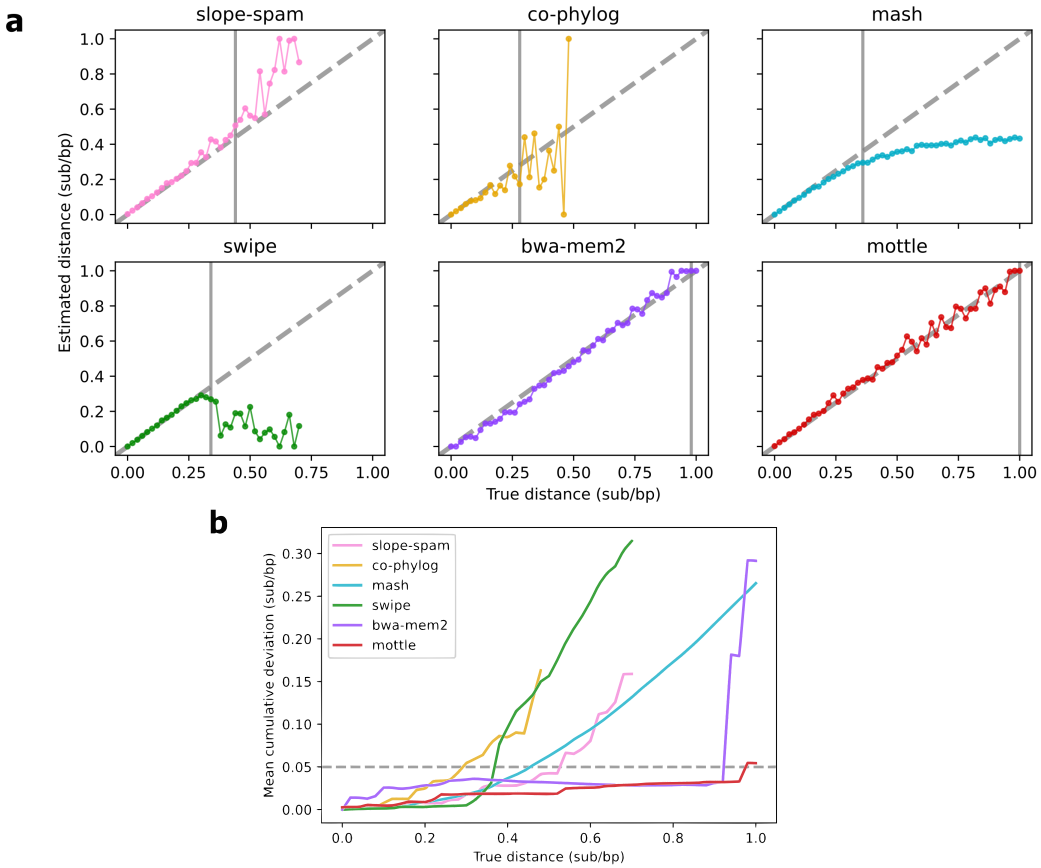


Figure 3.2 Accuracy of substitution distance prediction tools on a simple *in silico* substitution model of sequence evolution. **(a)** Program predictions vs true distance between sequences. Values are clipped to the range [0,1]. Vertical lines indicate the maximum stable distance. **(b)** Mean value of the cumulative deviation of each tool from the true distance. The maximum tolerable deviation is set to 0.05 sub/bp. The point at which curves cross tolerable levels defines the maximum stable distance. Curves are cut whenever NaN values are produced.

for many non-viral scenarios. These programs were not designed and tested with the high mutation rates observed in viruses in mind.

3.4.2. Known family alignments

Having established that Mottle could successfully be used to predict true distance over a large sequence divergence space – albeit in a simplified system - we next wanted to test Mottle against real viral sequences. However, it was important to maintain knowledge of the true distance between sequences. To do this, we employed the pre-aligned Rfam seed sequences of known RNA families, filtered to select viral sequences. From this, we calculated substitution distances from each alignment, to represent the true distance. To increase the length of these alignments and make them more comparable to a small RNA viral genome, alignments of similar divergence were concatenated to form sequences of >4kb in length. This created a more realistic dataset to test the six programs against. For all tools, predictive performance was degraded compared to the previous test (Fig. 3.3). For many programs, calculating distance was effective initially, but soon diverged from the true value. For most, distance was difficult to calculate beyond 0.25 substitutions per base (Fig. 3.3A). Co-Phylog, Swipe and Mash tended towards

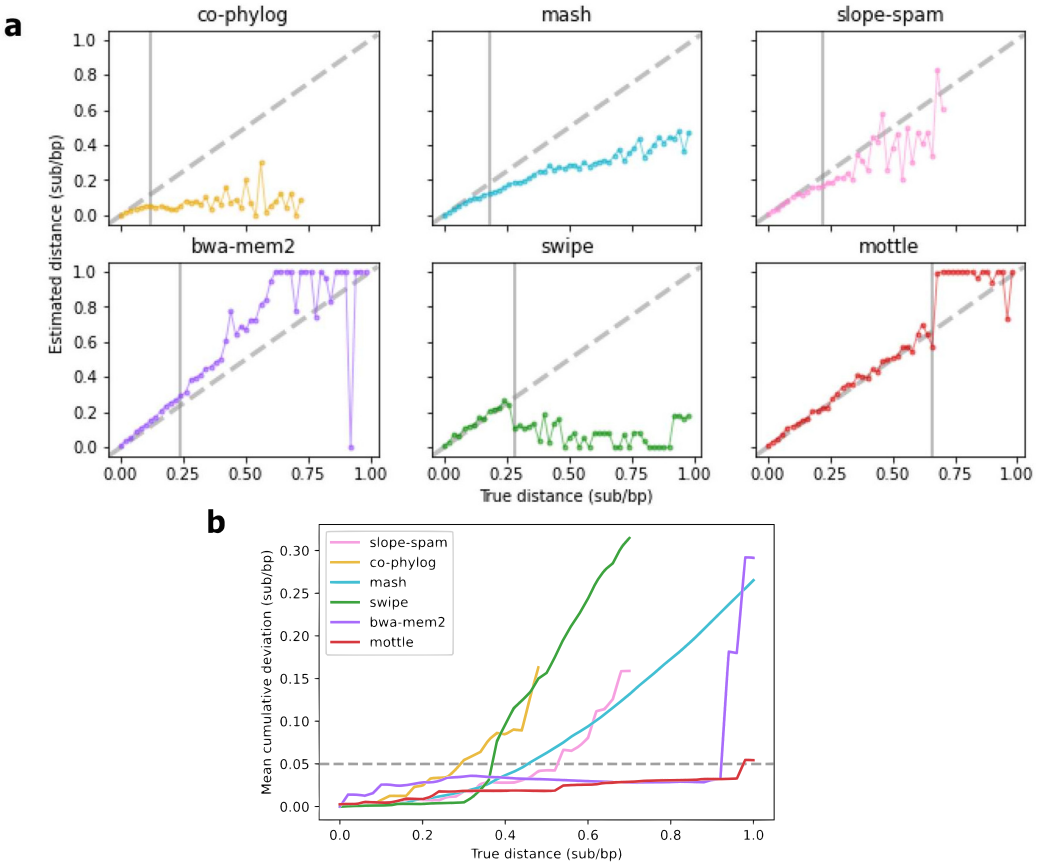


Figure 3.3 Accuracy of substitution distance prediction tools on a concatenated family alignment dataset. Formatted as Figure 3.2. **(a)** Program predictions vs true distance between sequences. **(b)** Mean cumulative deviation of each tool from the true distance.

underestimating sequence divergence from the true distance past this point, while Slope-Spam and Bwa-Mem2 displayed more erratic behaviour. By contrast, Mottle-map was able to track the true distance for longer, deviating only when reaching approximately 0.66 substitutions per base. All programs, except Mottle-map, had crossed the 5% cumulative deviation threshold by 0.3 substitutions per base (Fig. 3.3B, Table S2). In contrast to Fig. 3.2, all programs displayed similar deviation at low divergences. Taken together with the previous test, we conclude that Mottle is an effective tool for predicting true distance between highly divergent sequences, as may be found within viral populations.

3.4.3. Known genome taxonomies

We finally wished to test the performance of these programs in identifying the relationship between genomes in the known viral taxonomy. We assembled a test set of viral genomes composed of three different genomes at two different taxonomic ranks (i.e., genus, order, family, class). We used this dataset to assess how well each tool could identify out-group genomes as the taxonomic rankings were increased. Briefly, each program was required to calculate the distance between one reference genome and two others, one that shares a lower taxonomic ranking and another that only shares the rank above (out-group). A correct identification gives the out-group a higher distance to the reference. Unsurprisingly, most of the programs were able

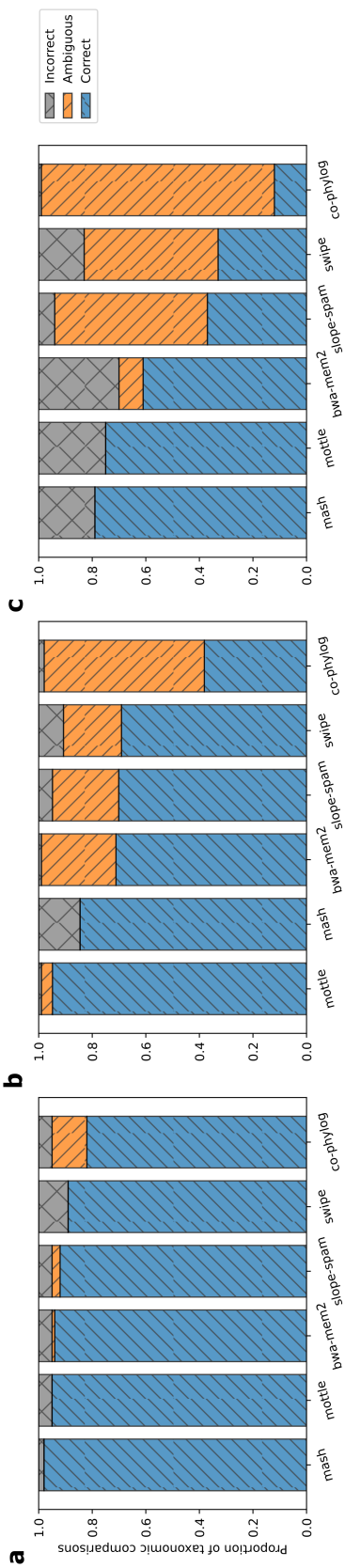


Figure 3.4 Accuracy of substitution distance prediction tools for identifying taxonomic out-group genomes. Proportion of assignments that were correct (out-group more distant than comparator genome), incorrect (out-group less distant), and ambiguous (identical distances) for each tool when comparing genomes in the same **(a)** Genus/Family, **(b)** Family/Order, **(c)** Order/Class. Predicted outputs for each test is listed in Appendix D.

to identify which genome shared the same Genus as the other, and which were only in the same Family (Fig. 3.4A). Mash performed the best under these conditions, followed by Mottle-map and then Bwa-Mem2. Co-Phylog, which was created with the genomes of cellular organisms in mind, did return several ambiguous results, perhaps a result of the instability of the output, even at low divergence, as seen in Fig. 3.4A. In the next test, the reference genome was compared to genomes in the same Family or Order (Fig. 3.4B). Here, all programs performed less well than in the previous test, with Mottle-map performing the best, followed by Mash and then Swipe. Again, Co-Phylog struggled to unambiguously place genomes correctly at this taxonomic distance. In the final test, the reference genome was compared with ones sharing an Order and Class (Fig. 3.4C). This test was far more challenging. Once again, Mash, Mottle-map and Bwa-Mem2 were the most effective, showing mainly correct placements with few ambiguous results. The other tools demonstrated high levels of ambiguity in this challenge, being unable to find any regions of homology to calculate a stable distance. Mottle, Mottle-map, Bwa-mem2, and Mash are therefore useful tools for placing test genomes within known taxonomies, even at large taxonomic distances. Full results for these benchmarks are available at Appendix D. A specific example of a prediction made by Mottle-map that was not made by other software is in finding similarity between Alstroemeria virus x and Potato virus M (0.5 sub/bp, Appendix D.3 entry 76), both in order Tymovirales, but falling out of bounds for Alstroemeria virus x and Lettuce chlorosis virus, only sharing class Alsuviricetes. Other software were unable to find similaritiy in either test, returning out of bounds or invalid values for both.

3.5. Conclusions

Pairwise substitution distance has a wide set of applications, from building phylogenies to creating reduced databases. Finding a novel approach in this field can find utilisations in any of these areas, giving opportunities for further refinement specific to certain applications. Mottle represents an additional tool in this endeavour. It performs as well as existing programs in correctly identifying taxonomic relationships, but comes with the added advantage of provided an accurate numerical estimation of genomic distance over a greater sequence divergence. A limitation of its use, though, is a reduced numerical accuracy at low divergences on genomes that behave similarly to a simple substitution model, i.e. where insertions and deletions are uncommon. Mottle may therefore be of particular interest to the study of viruses, viral relationships, and viral discovery platforms, where available sequences for reference may be sparse, sequence diversity is high, and insertions/deletions occur at a high rate. The algorithms behind Mottle can be applied using any Short-Read mapper, Gradient-Boosted Decision Tree generator, Gradient-Descent software, and Nearest-Neighbour finder. This means that any advancements in such software would give increased efficiency or accuracy to new implementations. Further extensions to the algorithm could include simultaneous multiple sequence distance calculation and isolating sub-alignments of continuous homology. In conclusion, Mottle is an invaluable, novel approach in substitution distance estimation, with significant performance benefits compared to other algorithms.

Chapter 4. Defining the Limits of Virus Detection Software

Summary: With current virus detection software the limits of their capabilities, and the factors that affect these capabilities, are not well quantified. For example, is a null result in viral detection analysis because no virus is present or because the sample contains a virus that is too far diverged from known reference sequences? Or, is a null result due to lack of read depth? And how does the size of the reference database affect pipeline performance? Here, we assess current detection tools (including traditional contig homology methods, less common assembly graph homology approaches, and a newer homology-independent software) for their capacity to detect viral genomes when query sequence reads diverge from sequences in the reference database, and the read depth is changed. The analysis reveals that two approaches, one based on homology detection (MMseqs2) and another using homology-independent detection (DeepVirFinder) perform robustly overall, but display important differences under specific conditions. For example, MMseqs2 was one of the worst performers at low read number but performed well with divergent sequences. Conversely, Mash Screen (homology-based detection) was able to perform well at low read numbers but did not tolerate family level divergence from the reference database. The work presented in this chapter identifies the limitations of current virus detection software, which will inform the analysis of plant viromes in the subsequent chapter.

4.1. Introduction

Accurately determining the presence or absence of viruses in a sample is a pressing concern for many parties, from the tracking of emerging diseases (Grubaugh et al., 2019) (Carroll et al., 2021) and the development of policy and regulation strategies (Terriau et al., 2021) (Bosch et al., 2018), to the profiling of microbial and host interactions (Matchado et al., 2021) (Albery et al., 2021) and the discovery of novel molecules for biotechnological applications (Varanda et al., 2021) (Roldão et al., 2011). Stating, with high confidence, that a sample is virus-free is a desirable outcome, but one that carries a risk of great harm (Hounsome et al., 2022).

Traditional methods for determining the presence of viruses and viroids in a sample can be divided into two major categories, targeted and untargeted. Targeted approaches require prior knowledge of what viral particles may be present in a sample. Cell culture had widely been considered the "gold standard" for microbiological detection, including for that of viruses (Hsiung, 1984), but is limited by which host cells can be cultivated, as well as the ability to inoculate and detect viral particles within them. Serological methods, such as enzyme linked immunosorbent assays, lateral flow immunoassays, and chemiluminescence immunoassays also

fall into this category. These require the selection of specific antibodies for their assay, introducing the possibility of missing unknown genomes. Similarly, PCR methods require the use of specific primers that target known sequences within sets of viral genomes. When a tissue shows symptoms of viral infection, but the all targeted tests return negative, this can create an issue for determining whether a novel viral genome is causing these symptoms or whether they are not due to a viral infection at all. In these situations, an untargeted approach must be taken. This traditionally involves the use of electron microcopy to visually confirm the presence of viral particles within the tissue, followed by morphological classification of these particles. Electron microscopy, though, is a labour-intensive process, and is not appropriate for the screening of large numbers of samples. For the purpose of viral detection, this creates a two-step protocol, where targeted assays are first performed on all samples, and follow-up electron microscopy is performed on selected samples as needed. This is likely to miss the breadth of possible viral genomes present within plants, where coinfection is known to occur, and the interaction between viral infections can influence the outcomes for the host plant (Miller, 2022). Genomic and transcriptomic sequencing had originally been considered a targeted approach, where the lack of a universal conserved sequence in viruses, such as the ribosomal DNA and RNA found in cellular organisms, required the design of PCR primers specific to each viral lineage. The use of randomly-primed PCR followed by shotgun sequencing were the first attempts to create a truly 'untargeted' virus nucleic acid detection technique (Clem et al., 2007). This was limited by low throughput, though, requiring stringent filtering of nucleic acids, whether by the use of nucleases to degrade non-encapsulated sequences, or size filtration to extract only viral particles. High-throughput sequencing has allowed the targeting of whole metagenomes and metatransciptomes of plant samples, creating the opportunity for assaying the entire viral content of a plant sample, the plant virome.

Just as traditional approaches contain diverse methodologies for virus detection, metagenomic protocols explore a large space of possible procedures. Chapter 1 detailed many components in the metagenomic detection of viruses, each with their own advantages and disadvantages. Though the multiple steps between sample collection and the analysis of sequencing data determine what types of viruses are targeted, pathogen agnostic methods have been developed to maximise the detection of all possible genomes (Simner et al., 2018). Specifically, the use of randomly primed high-throughput total RNA sequencing, with the depletion of abundant ribosomal RNA, is known to cast a wide net for different virus and viroid genomic organisations (Pecman et al., 2017). The computational steps taken after obtaining sequencing data, though, are less clear in their limitations. The standard approach towards this involves the application of *de novo* assembly, the attempted re-creation of the full-length set of sequences that the reads were generated from, followed by direct homology search against a reference dataset of known genomes or transcriptomes (Nooij et al., 2018). The limitations of this approach are widely known, correct assembly requires sufficient read depth, otherwise fractured contigs are generated (García-López et al., 2015), and homology search is limited in its sensitivity to viruses that are greatly divergent from known viruses included in reference databases (Ren et al., 2017).

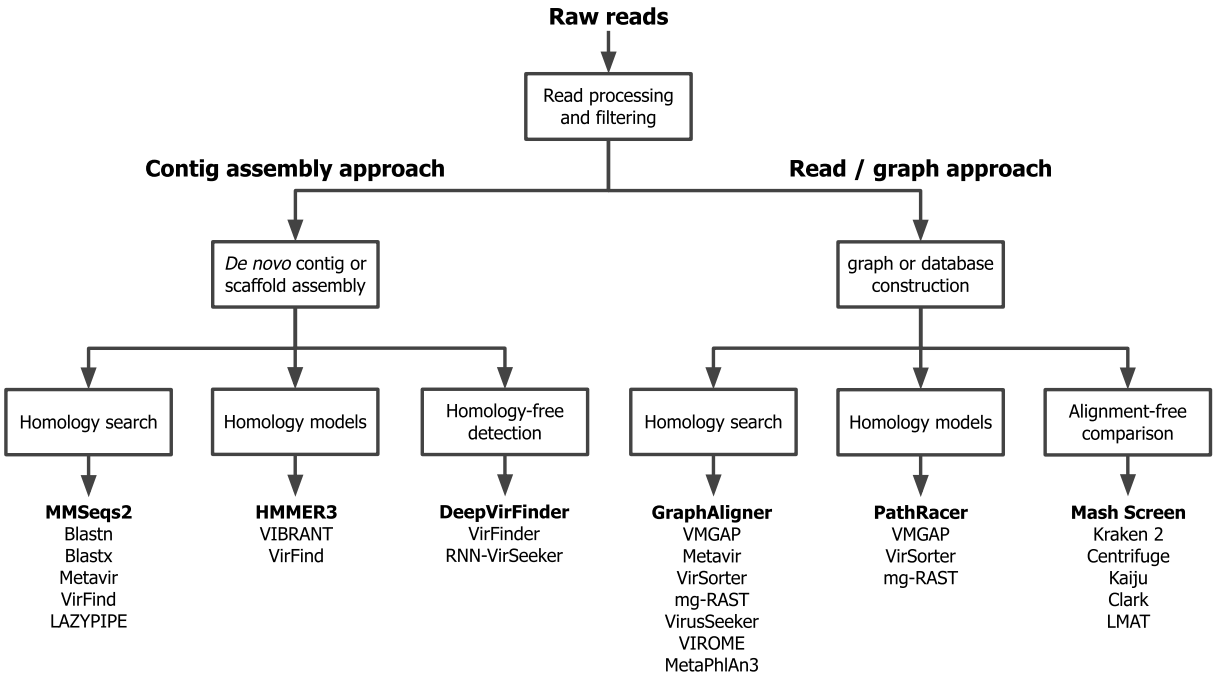


Figure 4.1 A selection of virus detection tools and pipelines, organised into broad categories based on their methodology. The first division is into approaches that utilise full contig assembly, and those that operate on the assembly graph or directly on the read set. Sub-categories for the contig-based group include direct homology search via alignment, the use of homology models such as Hidden Markov Models, and homology free approaches that leverage indirect knowledge, such as that from machine learning. The assembly-free category includes both homology search and homology model approaches, as well as alignment-free approaches. These approaches neither directly align reads/contigs to reference sequences nor align them to generated models, but instead compare other properties, such as k-mer distributions. Shown in bold are the selected tools for each category for this chapter.

This reliance on databases containing homologous genomes to those in a sample makes it difficult to count metagenomics as a truly 'untargeted' approach to viral detection. The fraction of sequencing datasets that are generated from viral sequences, but remain undetected, are known as the 'viral dark matter'.

A problem encountered in the metagenomic detection of viruses is that there are many factors that limit their efficacy. These include issues due to low depth and high divergence, but also includes the sparsity of viral taxonomy, i.e. some taxonomic groups have little representation within reference databases. In this chapter, we attempt to quantify the effects of these factors, and their combination, on the ability of software to correctly detect viral reads. We chose six bioinformatic tools that represent methodologically different approaches to viral sequence detection (Figure 4.1). We benchmark these software in their abilities to correctly label reads as viral, using a number of artificial and semi-artifical datasets. The aims of this chapter were to not only test the performance of the diverse set of viral detection software, but to define the limits of detection in terms of read depth, divergence, and reference database sparsity. That is, at which point does each factor begin to limit the ability to accurately detect viral sequences? And is there an interaction between these factors?

4.2. Materials and methods

The software used for virus detection in this chapter were MMseqs2 (Mirdita et al., 2021), HMMER3 (Wheeler and Eddy, 2013), DeepVirFinder (Ren et al., 2020), Mash Screen (Ondov et al., 2019), GraphAligner (Rautiainen and Marschall, 2020), PathRacer (Shlemov and Korobeynikov, 2019). The generation of viral labels for each of these is described in Chapter 2. Briefly, the input to each approach is a sequencing dataset, and a reference database for homology-based approaches, and the output is a set of viral scores for each read. For assembly-based approaches, read trimming and adapter removal is first performed by fastp (Chen et al., 2018b), followed by *de novo* assembly by rnaviralSPAdes (Meleshko et al., 2021). Additionally, read mapping to generated contigs is carried out by Bowtie2 (Langmead et al., 2019), to allow the transfer of contig-level scores to individual reads. For assembly-free approaches, adapter removal, but not read trimming, is first performed by dastp. De Bruijn graph construction is carried out where needed by the standalone spades-gbuilder program (Center for Algorithmic Biotechnology, 2022), otherwise, in the case of Mash Screen, k-mer database construction is done by the tool itself. Virus detection software is then executed on the prepared files.

4.2.1. Rfam concatenated genomes datasets

The first of our benchmarks used the Rfam concatenated genomes used in the previous chapter (Chapter 3), the creation of which is further described in Chapter 2. These artificial viral genomes were generated in pairs at a known substitution distance. Pairs were chosen that spanned a substitution distance between zero and one substitution per base pair (sub/bp), at

every 0.1 sub/bp, for a total of 11 genome pairs. For each pair, one genome was chosen as a reference genome and the other was used to generate query reads. Reads for each query genome were generated by InSilicoSeq (Gourlé et al., 2019), at either 1-times, 5-times, 25-times, or 125-times mean read depth, giving a total test space of 44 datasets. For each dataset, non-viral reads were also included from a *Nicotiana tabacum* plant sequencing run, NCBI accession SRR17544056. Bowtie2 was used with the `--very-sensitive` to map these the *Nicotiana tabacum* genome, Refseq accession GCF_000715135, with the first 1,000,000 reads being kept to use as non-viral background reads.

4.2.2. Filtered viral reads datasets

Our semi-artificial benchmark relied on further use of Bowtie2 read filtering. The non-viral background reads were the same as in the Rfam concatenated genomes datasets, but instead of *in silico* generated viral reads, we inserted reads from a Tobacco mosaic virus (TMV) sequencing dataset (NCBI accession SRR8073878) that mapped to the reference genome GCF_000854365.1. The first N reads were kept from this filtered dataset, where N was calculated to give a mean read depth of 1, 5, 25, or 125 times, via the formula

$$N = length \cdot depth$$

where length was the size of the reference TMV genome (6,395bp).

In addition to the read datasets themselves, we also generated the reference databases used for testing. As well as the *Nicotiana tabacum* reference genome GCF_000715135, we additionally included viral genomes with known taxonomic relationship to the Tobacco mosiac virus query. These were selected so that they only shared the taxonomic rank of interest and above, for example, Beet virus Q is in the same family as Tobacco mosiac virus, *Virgaviridae*, but the former belongs to the genus *Pomovirus*, and the latter *Tobamovirus*. Additionally, genomes were chosen as to maximise the taxonomic diversity within each rank, such as Drakaea virus A falling within the previously mentioned *Virgaviridae* family, but the genus *Goravirus*, different from Beet virus Q. Genomes used for each rank in testing is shown in Table 4.1. For each genome, substitution distance was calculated to Tobacco mosaic virus using *Mottle* with *Mottle-map*, using standard parameters: *ntrees*=100, *nleaves*=32, *learn_rate*=0.1, *subsamp*=1.0, *binpow*=64, *lean_mult*=0.001, *reltol*=1e-20, *maxiter*=100, *binthres*=0.75, *prior_size=10*.

Table 4.1 Viral genomes used to create reference databases.

Rank	1	2	3	4	5	6	7	8	9
Species	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus	Tobacco mosaic virus

Table 4.1 Viral genomes used to create reference databases (cont.)

Rank	1	2	3	4	5	6	7	8	9
Genus	Cucumber green mottle mosaic virus	Plasmopara viticola lesion associated tobamo-l...	Acidomyces richmon-densis tobamo-like virus 1	Brugmansia latent virus	Watermelon green mot-tle mosaic virus	Opuntia virus 2	Bottle gourd mottle virus	Piper chlorosis virus	Hoya chlorotic spot virus
Family	Apis virga-like virus	Drakaea virus A	Beet virus Q	Ligustrum mosaic virus	Peanut clump virus	Tobacco rattle virus	Soil-borne wheat mosaic virus	Hubei sed-iment vir-gavirus 1	Hubei sed-iment vir-gavirus 2
Order	Soybean ilarvirus I	Triticum polonicum clos-terovirus	Pteridovirus maydis	Solanum vio-lifolium ringspot virus	Lagenaria siceraria alphaen-dor-navirus	Tonate virus	Persimmon virus B	Sedum sarmen-tosum crinivirus	Cordyline virus 4
Class	Poinsettia mosaic virus	Rubivirus ruteetense	Currant virus A	Cassava Colom-bian symptom-less virus	Sanya benyvirus 1	Soybean-associated deltaflex-ivirus 1	Xylavirus EntGFV-2	Rocahepevirus ratti	Nudaurelia capensis beta virus
Phylum	Hepacivirus P	rice-associated noda-like virus 2	Luteovirus glycinis	Lake Sinai Virus SA2	Providence virus	Tombusvirus dianthi	Umbravirus patriniae	Aureusvirus cucumis	Machlomovirus zeae
Kingdom	Botrytis porri boty-birnavirus 1	Mivirus geno-vaense	Kummerowia striata our-miavirus	Apis hy-povirus 2	Taro-associated totivirus L	Leptosphaeria biglobosa mitovirus 3	Poecivirus A	Bovine astrovirus B76/HK	Karaka Okahu purepure emar-avirus

In tests where fewer than nine genomes were used, the first N genomes were selected, when N is the desired reference database size.

4.2.3. VIROMOCK challenge datasets

Datasets from the VIROMOCK challenge (Haegeman, 2021) were used for further benchmarking. These had a variety of properties and modifications to allow testing of a wide pool of scenarios (Tables 4.2 and 4.3). As host species, present viruses/viroids, and added viral genomes were known, we were able to give reads ground truth labels by mapping reads to these genomes with Bowtie2. The whole of the RefSeq viral genomes subset (Brister et al., 2015) was then used as reference databases.

4.3. Results

To determine the performance of virus detection tools in different scenarios, we used one synthetic and two semi-synthetic benchmarks. Our goal was to find the limits of these detection abilities, where performance is suddenly reduced. Each test used datasets where there was some

Table 4.2 Summary of VIROMOCK datasets used in this chapter.

Dataset	Researcher, institute, country	Dataset type	Plant species	Reads (bp)	Total number of reads
1	Kris De Jonghe, ILVO, BE	Semi-artificial	Citrus	2 x 150	21,703,434 (R1) 21,703,434 (R2)
2	Kris De Jonghe, ILVO, BE	Semi-artificial	Citrus	2 x 150	21,756,961 (R1) 21,756,961 (R2)
3	Marie Lefebvre, INRA, FR	Semi-artificial	Grapevine	2 x 150	24,526,416 (R1) 24,526,416 (R2)
4	Jean-Sébastien Reynard, AGS, CH	Semi-artificial	Grapevine	2 x 75	10,054,658 (R1) 10,054,658 (R2)
5	Denis Kutnjak, NIB, SI	Semi-artificial	Potato	1 x 50	31,277,475
6	Denis Kutnjak, NIB, SI	Semi-artificial	Potato	1 x 50	31,327,327
7	Paolo Margaria, DSMZ, DE	Real	Tobacco	2 x 301	1,904,369 (R1) 1,904,369 (R2)
8	Paolo Margaria, DSMZ, DE	Real	Chenopodium	2 x 301	65,177 (R1) 65,177 (R2)
9	Nihal Buzkan, UCDAVIS, USA	Real	Pistacio	2 x 151 (R1) 2 x 84 (R2)	5,259,903 (R1) 5,259,903 (R2)
10	Kristian Stevens, UCDAVIS, USA	Semi-artificial	Prunus	1 x 75	24,573,681

Table 4.3 Viral genomes known to be present, and modifications, in VIROMOCK datasets.

Dataset	Virus/Viroids already present	Modification	Challenge
1	CTV, CVEV, CEVd, CVd-III, HSVd	Addition of CTV	Different viral concentration (CTV strains)
2	CTV, CVEV, CEVd, CVd-III, HSVd	Addition of CTV	Mutation present in different frequencies (CTV haplotypes)
3	GRSPaV, GLRaV2, GRVFV, HSVd, GYSVd1	Removing of real viral reads	Different viral concentration (at the species level)
4	GRBV, GRSPaV, HSVd, GYSVd-1	Addition of GYSVd-2	Viroids with very similar sequence (GYSVd1 and GYSVd2)
5	PVY	Addition of PVY	Mix of recombinant and parental viral PVY strains
6	PVY	Addition of PVY	New PVY strain
7	TSWV	-	Complete genome + defective form of TSWV
8	PFBV + mitovirus	-	Cryptic mitovirus virus + low mitovirus concentration
9	PiVB	-	Concentration of different PiVB genomic segments
10	PBNSPaV	Addition of PPV	New PBNSPaV strain

knowledge of the genomes that were contained within them, whether it was completely artificial genomes and reads of the Rfam concatenated genomes test, the semi-artificial nature of filtered reads, or the spiked sequencing datasets of the VIROMOCK challenge. To assess performance of software at each point, we calculated the Area Under Precision Recall-Curves (AUPRC). Briefly, all reads in a dataset had ground truth labels of whether they were of viral or non-viral origin. Each virus detection tool was executed on the same dataset, and for homology-based tools the same reference database, returning viral scores for each read. Scores were then compared to ground truth labels, where at each possible threshold value, the precision (the proportion of values above the threshold that are true positives) and the recall (the proportion of all positives that are above the threshold) are calculated. The area under this curve, which can be thought of as a weighted average of precision as more true positives are included, is returned as the final performance score.

4.3.1. Rfam concatenated genomes

The first goal was to assess how each of the six pipelines performed in identifying viral reads, against a background of plant reads, with varying read depth and distance from the reference data. To do this, and to maintain control over the test, viral sequences were generated from the Rfam database by concatenating short sequences of known relation. This involved collecting previous alignments of RNA sequence families, calculating the pairwise substitution distances of the sequences contained within each family, and subsequently selecting pairs which match our desired substitution distance. These distances ranged from 0 to 1 substitutions per base pair. To generate longer sequences that are of equivalent length to viral genomes, paired short sequences of known relationship were concatenated to create longer sequence pairs. One sequence from each pair was used to generate a query sequence, while its cognate partner was used as the target in the reference database. To generate a background to mimic real-world sequencing data, sequencing reads from the tobacco plant was included in the data. In addition, mean read-depth of the query sequence was assessed. Sequencing data was simulated for each query sequence using InSilicoSeq. This allows the mean read depth to be set, generating data with varying levels of depth from 1 to 125-times. One times depth will almost certainly result in regions of the genome that are not represented in the read set, creating a great challenge for tools that rely on contiguity. Five times depth is considered the minimum level required by SPAdes for generating full length assemblies (Prjibelski et al., 2020), this should act as a 'threshold' level for assembly-based tools. 25 times depth is expected to be sufficient for most virus-length genomes, while 125 times is almost certain to generate full-length assemblies even for large sequences (Douglass et al., 2019).

The results of this analysis show that GraphAligner and MMseqs2 perform well across the majority of the challenge (Figure 4.2). GraphAligner shows significant advantage over MMseqs2 at very low read depth (i.e., 1-times). Mash Screen, HMMER3 and PathRacer also performed well at each read depth setting, though faced some difficulties in identifying the query sequence as viral when the distance between the two was > 0.7 substitutions per base pair (sub/bp).

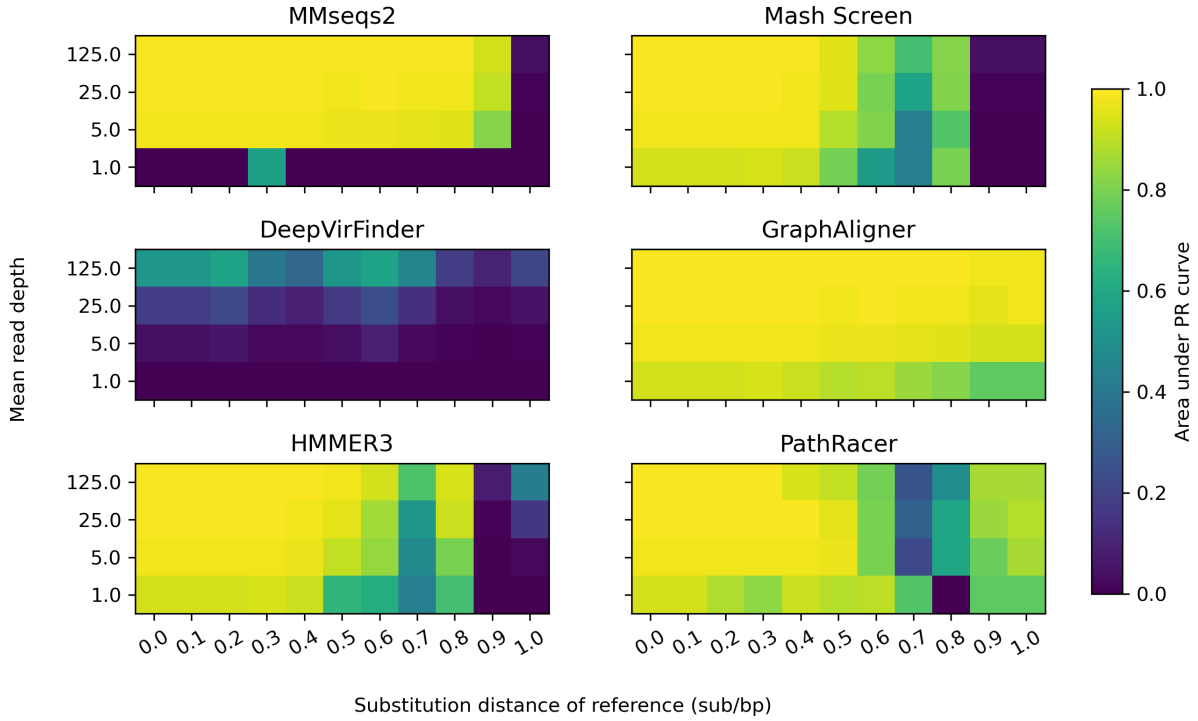


Figure 4.2 Comparison of viral discovery software performance at different substitution distance and read depth using Rfam concatenated artificial viral genomes. Each tool was challenged to identify reads of viral origin from an artificial viral genome set within tobacco background sequencing reads. Tools were tested using a reference data set with known substitution distance from the query sequence and a mean read depth of the query sequence of 1-, 5-, 25- or 125-times depth.

Interestingly, the success was not strictly dependent on the underlying methodology i.e., both assembly (MMseqs2, DeepVirFinder, HMMER3) and assembly-independent approaches (Mash Screen, GraphAligner and PathRacer) showed a variety of behaviours. DeepVirFinder performed especially poorly on this benchmark, only reaching moderate performance when at the highest (125-times) depth and below 0.8 sub/bp. This initial assessment of software capability, however, has some limitations that may influence interpretation of the results. DeepVirFinder for example, has been trained against viral genome sequences and the lack of success in this test, may be attributed to the artificial nature of the query sequence. Likewise, the favourable performance of GraphAligner, and to a lesser extent PathRacer, may be due to an artificial advantage in this test. The use of RNA family sequences, which lack codon structure, could separate generated reads from host reads in the assembly graph, which would make it less likely for graph-based approaches to mischaracterise host sequences as viral.

4.3.2. Filtered reads

Having assessed each pipeline under completely controlled conditions (i.e. using concatenated Rfam sequences) the next test was against real genomes. To do this, sequences were obtained from the NCBI sequence read archive and filtered to obtain a test data set comprising both host and viral sequences. Each pipeline was then assessed for their ability to label viral reads (and non-viral reads) correctly. As in the previous section, the conditions for the assessment varied

both by taxonomic distance from the reference data set (here, varying from species to kingdom) and mean read depth of the query sequence (from 1- to 125-times). In this test, the size of the reference data set was also varied, using one, three, or nine complete viral genomes within the reference. For homology-model tools (HMMER3 and PathRacer), Pfam and Rfam (extracted from Covariance Model files) Hidden Markov Models were aligned to the reference dataset. For each aligned model, a new model was generated that included only the reference alignments.

This test proved more challenging, with the performance of all pipelines being reduced compared to the previous test (Figure 4.3). Again, MMseqs2 and PathRacer proved some of the most robust methods, providing mean read depth was 25 or higher, though both were unable to correctly identify query viral sequences at the highest distances (shared phylum and kingdom) from the reference. Interestingly, GraphAligner, which has performed very well in the former test, was unable to identify viral sequences at distances beyond the genus, behaving more similarly to Mash Screen, which also showed a low performance beyond the genus level. DeepVirFinder by contrast, was highly effective at identifying viral sequences even at large taxonomic distances. This is to be expected, as it does not rely on the reference sequences at all, instead identifying viruses using a deep learning approach. DeepVirFinder however, does require good read depth, and is very sensitive to reductions in mean read depth. This is likely due to the fact that it is heavily reliant on the assembly of a high quality draft genome as part of the process.

HMMER3, and to a lesser extent Pathracer, showed a discrete probabilistic reduction rather than a smooth tapering off in performance when below ten times depth (for both) or above family-level (for HMMER3). This is to be somewhat expected - both tools used sequence models generated from the reference database, so only homology present between the query genome and generated models are captured. This can lead to a binary outcome when it comes to viral read detection - either a model, or models, were generated that captured homology between the genomes, or not. This seemed to affect HMMER3 more than Pathracer, where the former is assembly-based, and the latter is graph-based. Since contig assembly in many cases can also generate gaps of homology, i.e. reads that were left unassembled, this likely compounded the sparsity of homology.

Reference database size showed a limited effect on the final outcome in most instances. The largest effect was seen for MMseqs2, which saw its performance greatly reduced at the Order level. Additionally, Mash Screen and GraphAligner both saw some reduction in performance at the species level. HMMER3 saw a stochastic reduction in performance beyond the species level, though this was most constrained to the Order level and above. Surprisingly, PathRacer was largely unaffected, despite a reliance on homology models. As it operates on read graphs, this may allow it to capture all present regions of homology, even for that of a single reference genome. DeepVirFinder, not being reliant on the reference database, was completely unaffected.

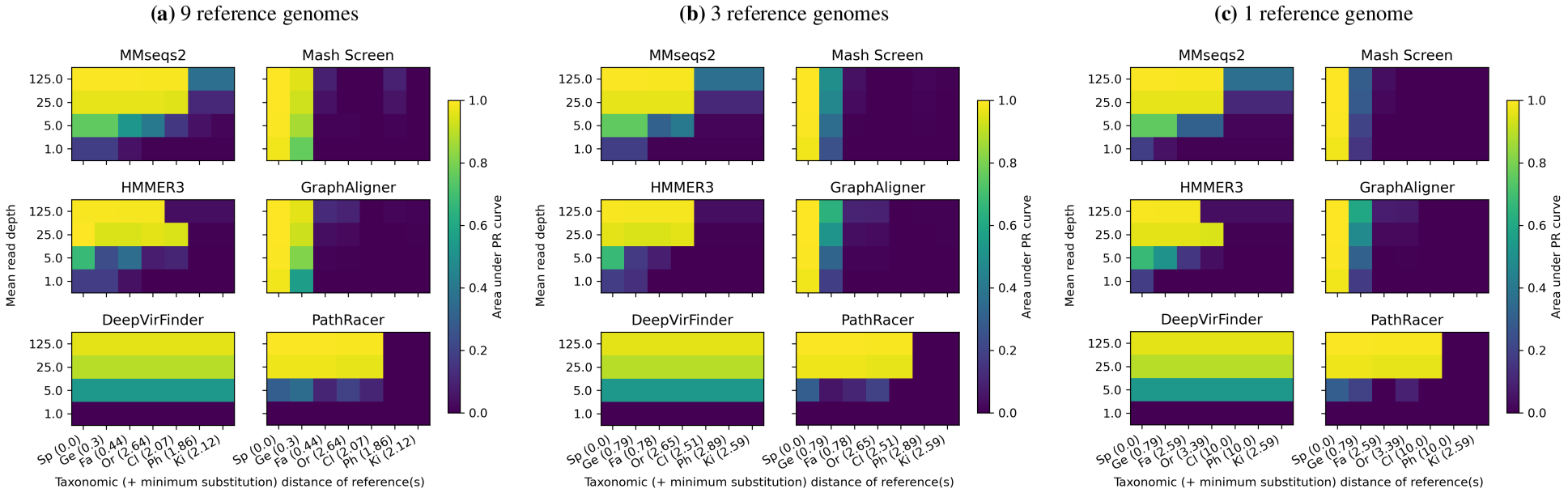


Figure 4.3 Comparison of viral discovery software performance at different taxonomic distance, read depth, and reference database size, using NCBI taxonomy relationships. Each tool was challenged to identify reads of viral origin from a Tobacco Mosiac Virus sequencing dataset set within tobacco background sequencing reads. Reads were aligned to viral and host genomes, and successfully mapped reads were extracted to generate a semi-artificial dataset. Viral reads were subsetted to obtain mean read depth of the query sequence of 1-, 5-, 25- or 125-times. Reference genomes were selected that had certain taxonomic relationships to the query genomes, e.g. falling within the same family but not genus. Additionally, the number of reference genomes used was either nine, three, or one genome.

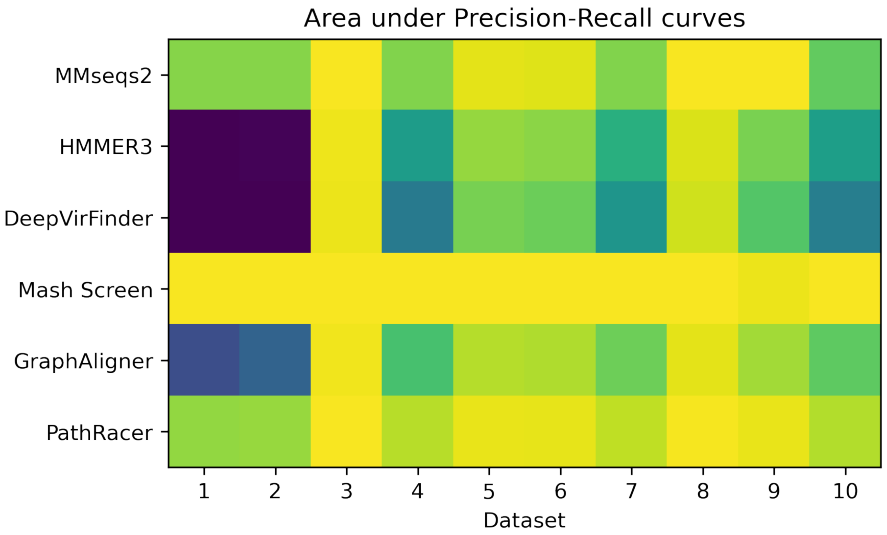


Figure 4.4 Performance of virus detection software in the detection of viral reads in the VIROMOCK challenge. Reads were mapped to viral genomes known to be present in each dataset, which were then used as ground truths in the calculation of Area Under Precision-Recall Curves (AUPRC) for each tool.

4.3.3. VIROMOCK challenge

In the following test, the performance of each pipeline was assessed using a community standard. Specifically, we used data sets available as part of the VIROMOCK challenge (Waite et al., 2022). This consisted of the ten semi-artificial and real datasets (Datasets 1-10). In the semi-aritificial datasets, a background of sequencing reads are spiked with *in silico* generated reads, similar to our taxonomy-based benchmark. Additionally, viral genomes that are known to be present in the background reads have been listed (Table 4.3). This knowledge was used to map reads from each dataset to the known viruses, creating a ground truth of viral reads. This was also done for each host genomes, generating known labels for non-viral reads. Unmapped reads were included, but were left without a ground truth value. The performance of the six tools on these datasets is summarised in Figure 4.4.

There was a general trend in terms of the performance of tools on this dataset. Mash Screen showed a very high performance on all datasets, reflecting the species-specific knowledge of the viruses present within the datasets. Performance in these challenges also did not follow software methodology, with MMseqs2 and Pathracer, the former assembly and homology search based and the latter graph and homology model based, showing the next greatest performance.

Some datasets, i.e. 3 and 8, all tools showed a high level of performance. The former (dataset 3) excluded real viral reads, so that tools only had to detect *in silico* generated reads, which they exhibited great performance in. Dataset 8, on the other hand, did not include any artificial reads. It was also the smallest dataset, at 65,177 paired reads, and sequencing was performed with a read size of 301bp (Table 4.2). These longer reads, with likely fewer background genomes, were sufficient to guarantee correct assembly and therefore performance for assembly-based tools, whereas read- and graph- based tools were not limited by read count.

In other datasets, virus detection software showed a much higher deviation in performance, especially in datasets 1 and 2. HMMER3 and DeepVirFinder, both assembly-based, showed especially poor performance on both, whereas Mash Screen, PathRacer, and MMseqs2 showed good performance, with GraphAligner in-between. These datasets utilised artificial Citrus tristeza virus in a Citrus background. This virus is known to have a large genome for an RNA virus, at 20kb. Dataset 1 used different strains of this virus, while dataset 2 introduced mutations at different frequencies. This diversity may have interfered with the assembly of the large genome, creating fractured contigs. These shorter contigs would create difficulty for both tools, similar to the very low depth seen in Figure 4.3, where gaps in homologous regions would cause sequence models to miss their targets in HMMER3, and shorter contigs missing the viral genome structure that DeepVirFinder is trained on.

4.3.4. Runtime statistics

Although the focus of our tests were to find the limits of detection, the physical resource usage of each program could not be ignored. Total CPU usage is shown in Figure 4.5a, and maximum memory using in Figure 4.5b.

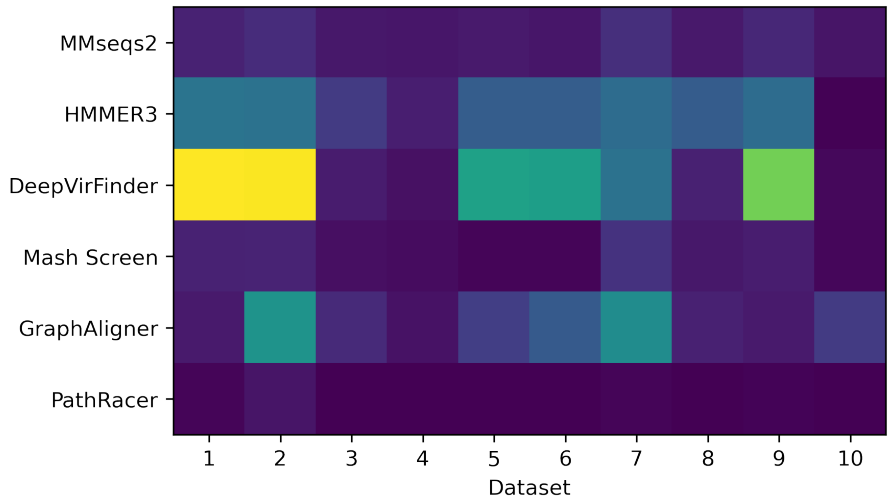
4.3.5. Determining optimal thresholds

While the AUPR allows us to test the overall performance of virus detection software when ground truth labels are known, this is not applicable to real sequencing datasets, where a discrete threshold is required for including an output as a positive 'hit'. Each tool uses different methodologies for calculating output scores, so to allow unbiased comparison between tools in the next chapter, we algorithmically determined optimal thresholds using the results of the filtered reads benchmark. At each threshold value, we calculated F1 scores, defined as the harmonic mean of precision and recall, or in a simplified form as:

$$\frac{2 \cdot precision \cdot recall}{precision + recall}$$

Thresholds were then chosen so that, for any read depth, taxonomic distance, and reference database size, that showed an AUPRC above 0.1, their maximum F1 score was above the threshold. This allows us to make sure hits are above threshold, even near the limits of detection. F1 scores at varying thresholds is shown in Figure 4.6. For most tools, increasing thresholds decreased F1 score. Stricter thresholds reduce the number of true positives included as hits, so if there is no concurrent reduction of false positives that balance this out, F1 score decreases. This was the case for all tools bar GraphAligner, which showed increasing F1 scores until a peak. This shows a significant presence of high-scoring non-viral reads in its output. The final calculated thresholds are as follows, (MMseqs2: 34.0, HMMER3: 14.2, DeepVirFinder: 3.4, Mash Screen: 1.0, GraphAligner: 4.4, PathRacer: 4.4.

(a) Total CPU time.



(b) Maximum memory usage.

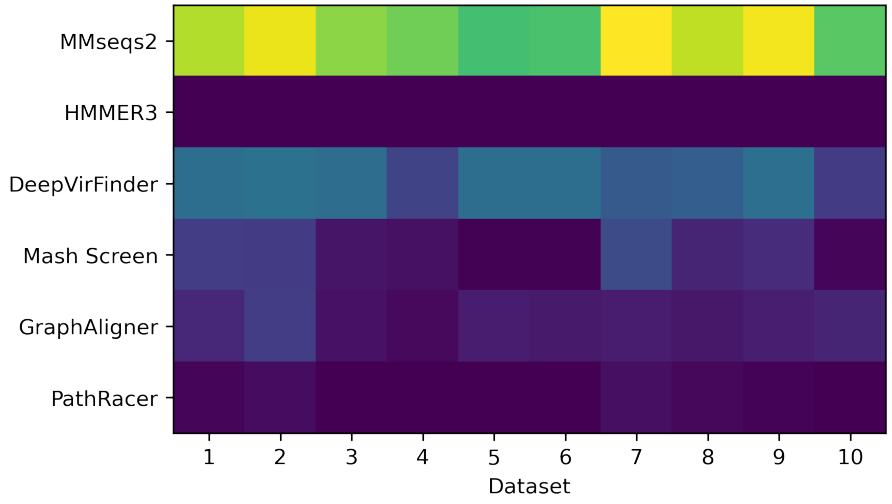


Figure 4.5 Runtime statistics of virus detection software when executed on VIROMOCK datasets. Showing (**a**) total CPU runtime and (**b**) peak memory usage.

4.4. Conclusions

The performance of virus detection tools showed great variation within each of the tests, and between them. On artificial genomes, All tools apart from DeepVirFinder showed good performance on most of the test space, with MMseqs2 being mainly limited by read depth, and with HMMER3 and Mash screen being mainly limited by reference dataset divergence. There was no obvious split between assembly and read/graph based tools. On the other hand, the filtered read test showed different patterns, with Mash Screen and GraphAligner being able to tolerate low depth, DeepVirFinder not being limited by high divergence. MMseqs2 and PathRacer showed a balance between the two. Crucially, no single tool was able to cover the whole possibility space of limiting factors, showing the importance of not relying on a single approach.

Though we have shown that all current metagenomic virus detection have limitation when it comes to artificial datasets, analysis of their performance on real sequencing data was limited to

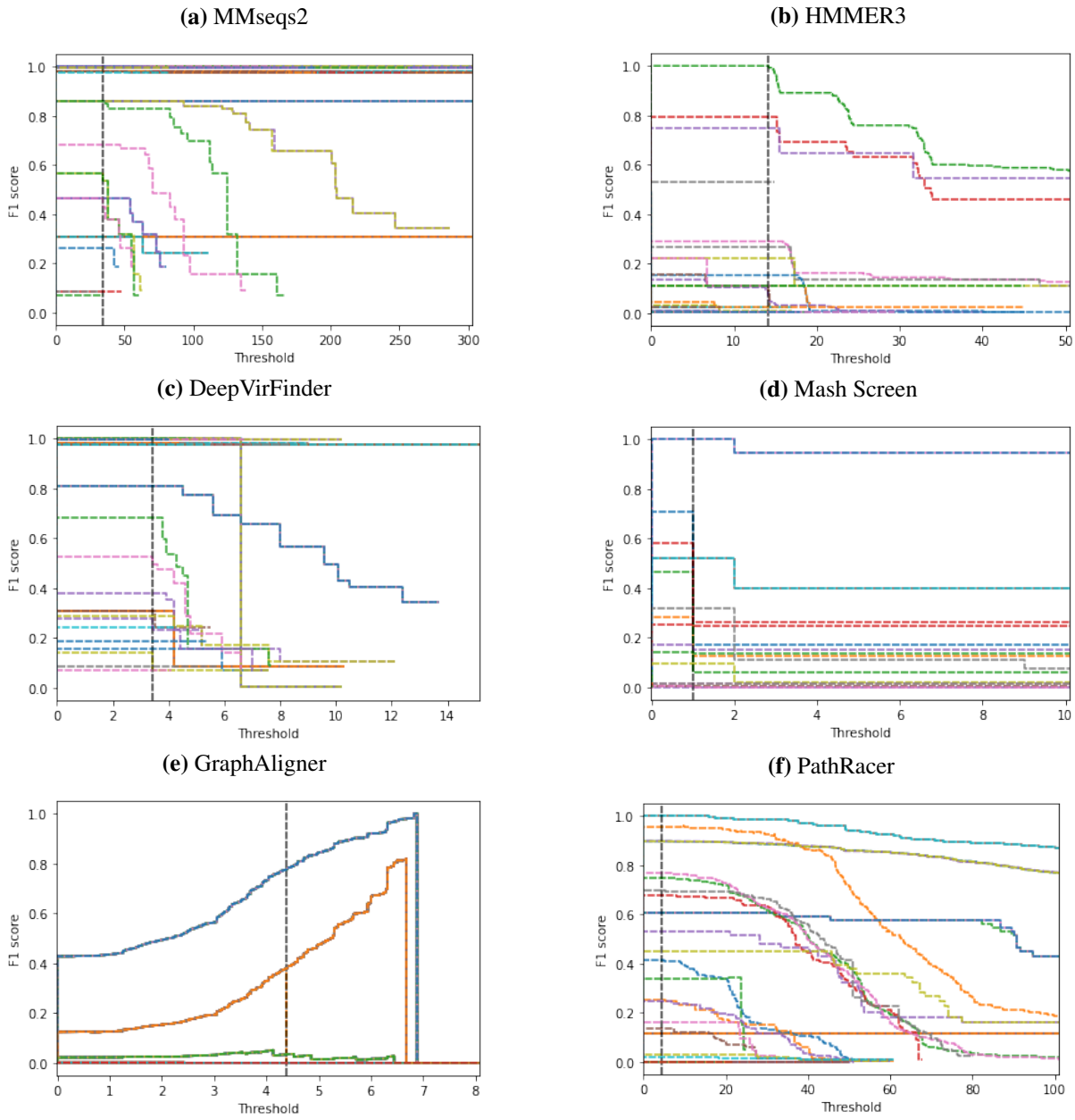


Figure 4.6 F1 scores of viral detection tools when threshold of detection was varied. Vertical lines indicate optimal threshold in terms of including F1 maximums at all parameters near the limits of detection.

the quantitative VIROMOCK tests. In the next chapter we analyse their qualitative performance on real sequencing datasets.

Chapter 5. Qualitative Analysis of Viral Detection Software

Summary: Analysis of the whole viral fraction of plant metagenomic datasets, plant viromics, relies on the ability to detect the presence of viral genomes in a complex background of host sequences, microorganisms, insect remains, and environmental genetic contaminants. Within this murky pool, lie putative viral reads, those that may be of viral origin, but we may not be able to detect - the viral dark matter. Metagenomic software promise the ability to pick apart these entangled reads, neatly separating them to their originating genome. This process, though, is often not so simple, with different tools applying the label of 'virus' in different ways. Whether a read is of viral origin then, may depend the choice of software. In this chapter, we apply the programs characterised in Chapter 4 to previously seen and novel metagenomic datasets. We compare and contrast their assignment of reads as being of viral origin, analyse their level of agreement when broken down by genome, and evaluate their ability to reach a unified conclusion. We find that there is indeed much disagreement between tools in terms of which reads are considered to be of viral origin, but a surprising amount of agreement in which genomes are present, often returning similar read numbers. This is most striking for the only homology-free software used in these tests, DeepVirFinder, which is highly prolific in the numbers of putative viral reads it detects, but often returns very similar numbers of reads to homology-based tools for viral genomes that we would expect to be present. This leads us to support the use of a mixed approach - the conclusions that can be drawn using a homology-based tool together with a homology-free tool is greater than the sum of its parts.

5.1. Introduction

The study of viromes is the study of viral and viroid metagenomes associated with an environment, such as oceans (Hurwitz and Sullivan, 2013) and soils (Paez-Espino et al., 2016), or associated with an organism, such as humans (Wylie et al., 2012) and nematodes (Vieira et al., 2022). Analysis of plant-associated viromes specifically has been employed for novel virus discovery (Yang et al., 2022), biosecurity (Wylie et al., 2019), spatial virus distribution (Cao et al., 2021), and network analysis to uncover agroecological interactions (Alcalá-Briseño et al., 2020). The conclusions drawn in these studies are reliant on the ability to accurately detect the presence of viral genomes in their datasets. Analyses that find differing plant viromes in different settings would lead to differing conclusions, and possibly differing biosecurity or policy decisions. The influence of the choice of sample preparation method and sequencing platform on plant virome conclusions is well documented (Gaafar and Ziebell, 2020; Pecman et al., 2022, 2017), but the influence of virus detection software used for their analysis has had

less attention outside of *in silico* benchmarks. Bester et al. (Bester et al., 2021) mainly investigated the influence of sample preparation and sequencing platform on plant virome characterisation, but also compared the use of *de novo* assembly and homology search versus direct read mapping. Read mapping detected the full complement of expected viruses and viroids regardless of preparation and sequencing approaches, but the assembly approach missed some viroid genomes in some replicates of Illumina sequencing datasets, and missed many virus and viroid genomes in the shorter-read Ion Torrent sequencing dataset. Additional analyses on non-plant metagenomes have been carried out by Ye et al. (2019) and de Vries et al. (2021). The former compared species abundance profiles between tools on a metagenomic sample with a predefined composition, finding that k-mer based tools were better able to detect correct read numbers compared to alignment based tools. The latter, on the other hand, used clinical samples with PCR-confirmed viral genomes. This was used to test the ability of many tools, split into assembly and non-assembly, for their taxonomic accuracy in the detection of genomes, comparing whether detection were accurate to the species, serotype, strain, or isolate level. They found no major difference between assembly-based and read-based tools in their performance, though a greater proportion of assembly-based tools showed a high sensitivity.

The detection of viral genomes in a plant sample, regardless of technique, is generally limited by three main factors: rates of mutation, low viral titres, and sparse taxonomies. The numerical extent to which these factors influence our ability to detect the presence of a virus was characterised in Chapter 4, using various synthetic benchmarks. There was a clear difference in the behaviour of viral detection software at these limiting factors, where contig assembly based tools (MMseqs2, HMMER3, and DeepVirFinder) generally showed the greater ability to detect highly divergent viruses, whereas read and graph based tools (Mash Screen, GrapAligner, and PathRacer) tended toward a greater performance at low viral titres. This was subverted, however, by PathRacer, which showed the greatest performance at high divergence of any homology-based tool. How, and whether, these differing behaviours at the limits of detection influence their outputs when executed on real datasets is still unclear. For example, does using different software for virus detection bias conclusions? Additionally, is there a qualitative difference in what these methodologically diverse tools consider a positive hit?

There are many aspects of viral metagenomics that can be used to compare the outputs of metagenomics software, such as quality of genome assembly, species abundance estimation, and differentiation between strains. These tests generally limit the scope of which software can be tested, such as assembly quality not being applicable for tools that do not generate assemblies, and species abundance not being an appropriate test for non-homology tools. In this chapter, we expand on the simple metric developed in Chapter 4, of calculating read-level scores for each tool. By comparing which reads are considered to be of viral origin by each tool, we can find commonalities and differences in their outputs in a way that is agnostic to exactly methodology of each tool, i.e., whether a tool relies on assembly, non-assembly, homology, or non-homology approaches. This allows us to tackle how, and whether, there is a difference in the output of diverse virus-detection software, and whether they can be used together to reinforce conclusions.

5.2. Materials and Methods

In this chapter, we apply the viral detection tools characterised in chapter 4 on plant leaf whole-transcriptome RNA sequencing (RNA-seq) datasets. Six samples were collected (5.2.1), then prepared and sequenced (5.2.2), to generate read datasets. Which reads were labelled as being of viral origin were recorded for each tool, and compared across tools to find similarities and differences in their output (5.2.4).

5.2.1. Sample collection

Samples used for generating datasets were obtained by partners at Fera Science Ltd. Six samples were collected from three separate surveys. The first of these was the Pea coinfection dataset, which was generated from commercial *Pisum sativum* samples as part of Fera Crop Health services, where samples from symptomatic plants are sent in by clients for viral diagnostics. This sample had been tested by Enzyme Linked Immunosorbent Assay, and had a confirmed presence of multiple isolates of Turnip yellows virus. It was then freeze-dried before being subjected to preparation and sequencing (5.2.2).

The next set was obtained from the CaLiber survey (CALIBER, 2023), an ongoing effort to research the bacterial plant pathogen *Candidatus* Liberibacter *solanacearum*, as well as its plant and insect hosts, within the UK. Two CaLiber samples were used in this chapter, which were taken from leaves of *Heracleum sphondylium* (CaLiber Hogweed) and *Urtica dioica* (CaLiber Nettle), which were symptomatic of viral infection. These samples were then directly subjected to preparation and sequencing.

The final three datasets used were generated from Pea bulk sequencing samples (Fowkes Pea 14, Fowkes Pea 20, Fowkes Pea 15). These samples had been previously analysed and published in (Fowkes et al., 2021). Briefly, samples of *Pisum sativum* crops from across the United Kingdom were collected by staff from the Processors and Growers Research Organisation (PGRO). Within each site, 121 leaf-top samples were taken across a 10 metre by 10 metre grid, 120 of which were then sent to FERA viral diagnostics. Samples were prepared and sequenced to detect candidate virus. Sequencing data was analysed using an in-house tool, Angua3 (McGreig, 2022), to detect putative viral genomes, which were then confirmed by Reverse Transcription Polymerase Chain Reaction (RT-PCR) and real-time RT-PCR. From these datasets, three were selected by partners at Fera, in which the presence of multiple virus and viroid genomes was confirmed by RT-PCR. Knowledge of which datasets contained which confirmed viruses was not given until after our own analysis had concluded.

5.2.2. Sample preparation and sequencing

Sample preparation and sequencing for all samples was carried out by staff at Fera Science Ltd. Methodology was as detailed in (Fowkes et al., 2021). Briefly, a cork borer was used to collect material from a sample leaf, where in the case of bulk sequencing datasets this was done for

each leaf separately. RNA was extracted from collected material by Qiagen RNeasy mini kit (Qiagen, 2023), following manufacturers' instructions, and including a DNase step. The extract was then processed using the Illumina TruSeq Stranded Total RNA with Ribo-Zero Plant (Illumina, 2023), first depleting the abundant ribosomal RNA from the host plant (and other organisms), then generating dual unique indexed libraries. For bulk sequencing, libraries were pooled together at this stage. Single and bulk libraries were then diluted to 10 pM, combined with 5% PhiX library for positive control, and sequenced on an Illumina MiSeq using a 600 cycle V3 kit. Summary statistics for resulting datasets are shown in table 5.1.

5.2.3. Dataset processing

Each dataset was processed following the same methodology used in the previous chapter (detailed in Chapter 2.4). Briefly, for assembly-based tools, reads were subjected to adapter removal and trimming followed by *de novo* assembly. For non-assembly approaches, adapter removal was carried out without further processing.

5.2.4. Comparative analysis of viral detection software

Viral detection tools were applied to each dataset following the methodology detailed in Chapter 4.2. The six software used were MMseqs2 (Mirdita et al., 2021), HMMER3 (Wheeler and Eddy, 2013), DeepVirFinder (Ren et al., 2020), Mash Screen (Ondov et al., 2019), GraphAligner (Rautiainen and Marschall, 2020), and PathRacer (Shlemov and Korobeynikov, 2019). The reference database used for homology-based tools in this chapter was the NCBI RefSeq genomes database (O'Leary et al., 2016). To allow comparison between homology and non-homolgy tools, only viral score of each read is kept from their outputs. These scores were thresholded using optimal scores identified in 4 (MMseqs2: 34.0, HMMER3: 14.2, DeepVirFinder: 3.4, Mash Screen: 1.0, GraphAligner: 4.4, PathRacer: 4.4) to produce binary labels, i.e. whether the read is identified as viral or non-viral. The resulting labels could then be compared between tools on a read-by-read basis.

We first calculated, for each read, the number of tools that had returned a positive viral label, which we termed the degree of overlap. Stacked bar plots for each dataset, for the number of reads that had each degree of overlap, were visualised with Matplotlib (Hunter, 2007). We then calculated pairwise Szymkiewicz–Simpson overlap coefficients between tools using the python package NumPy (Harris et al., 2020), where the putative viral reads of each tool were treated as overlapping sets. Visualisation was also carried out with Matplotlib. To allow higher-degree overlap computation and visualisation, we applied the UpSetPlot python package (Nothman, 2023). This generated an intersection matrix that contained counts for the number of reads in each intersection, and visualised each as a membership matrix together with a count bar plot, as in (Lex et al., 2014).
To associate putative viral reads with viral genomes, we used a map and propagate approach. First, we aligned all adapter-removed reads with a degree of overlap above one, i.e. labelled as a

Table 5.1 Summary statistics for raw datasets used in this chapter.

Dataset	Read count	Total size (nt)	Min len (nt)	Avg len (nt)	Max len (nt)	Q1	Q2	Q3	N50	Q20 (%)	Q30 (%)
Pea coinfection R1	333,737	83,743,503	32	250.9	301	206	284	301	300	93.12	86.86
Pea coinfection R2	333,737	84,906,321	32	254.4	301	209	299	301	300	84.96	74.78
CaLiber Hogweed R1	647,190	153,612,578	32	237.4	301	192	248	300	273	97.14	92.21
CaLiber Hogweed R2	647,190	157,665,029	32	243.6	301	193	268	301	300	86.69	76.66
CaLiber Nettle R1	1,170,935	293,665,875	32	250.8	301	210	271	300	289	95.83	90.05
CaLiber Nettle R2	1,170,935	298,333,120	32	254.8	301	213	288	301	300	88.23	78.43
Fowkes Pea 14 R1	710,605	167,935,665	32	236.3	301	179	252	300	291	96.84	90.20
Fowkes Pea 14 R2	710,605	169,018,293	32	237.9	301	179	256	301	300	89.93	78.19
Fowkes Pea 20 R1	842,649	151,681,705	32	180.0	301	140	177	208	189	89.25	81.57
Fowkes Pea 20 R2	842,649	154,233,128	32	183.0	301	139	155	238	174	73.22	65.70
Fowkes Pea 15 R1	1,034,408	223,626,458	32	216.2	301	168	208	276	235	97.65	92.78
Fowkes Pea 15 R2	1,034,408	224,348,979	32	216.9	301	168	209	281	236	94.51	86.65

Total size: Sum of read lengths, in nucleotides, of database. Min len: Minimum read length. Avg len: Average read length. Max len: Maximum read length. Q1, Q2, Q3: First, second (median), and third quartiles of sequence length, respectively. N50: When reads are sorted from shortest to longest in length, this is the length of read (rounded to shorter) so that the sum of read lengths above and below are approximately equal, i.e. median read length when weighted by length. Q20: Percentage of bases with quality score greater than twenty. Q30: Percentage of bases with quality score greater than thirty. R1: Forward reads of dataset. R2: Reverse reads of dataset.

viral read by at least one tool, to the RefSeq viral genomes subset (Brister et al., 2015). This was accomplished using Bowtie2 (Langmead et al., 2019) alignment with parameter `--very-sensitive-local`, which gave the greatest sensitivity to any homology to a viral genome. Identities for each read mapping were calculated using msamtools (Arumugam Lab, 2023). As some reads may be missed if they fall within a highly divergent region of a viral genome, reads were also mapped to contigs generated by rnaviralSPAdes (Meleshko et al., 2021). Reads that co-assembled, that is, mapped to the same assmbled contigs, then had their genome mappings propagated to reads that did not have a mapping, where the highest identity mapping was propagated. Final tables were generated using pandas (The pandas development team, 2020).

5.3. Results

5.3.1. Degree of overlap between virus detection software

To broadly characterise the similarities and differences in the outputs of the set of software on each dataset, we first calculated the overlap of reads assigned as viral between all tools. For each read, the number of tools for which it passes the score threshold is totalled. This we termed the degree of overlap. As each tool behaved differently depending on limiting factors in Chapter 4, we expected much heterogeneity in their assignments. Specifically, we expected some viral genomes to be present at low titres, especially in the Fowkes Pea datasets, which had pooled reads. This could pose a problem for MMseqs2, HMMER3, Pathracer, and especially DeepVirFinder, which all showed degraded performance at low read depths. Results are summarised in Fig. 5.1. For all datasets, the majority of reads were not labelled as viral by any tool. This is consistent with host plant RNA being the most abundant target of reads, even at high viral loads. In the majority of datasets, the most common degree of overlap is one. This indicates that at least one tool generates many uncorroborated assignments. Two and three degree overlaps were uncommon, outside the CaLiber Nettle dataset. This suggests a divide between the assignments of subsets of software on this dataset. Degree six overlaps, i.e. where all tools agreed on the viral status of a read, did not feature significantly in many of the datasets, with the most visible faction present in the Pea coinfection dataset. This indicates a core set of unambiguous viral reads in this dataset. The lack of unanimity by viral detection software on most datasets shows there are many reads for which it is not clear whether they are viral in origin, and that there would be a difference in labelling based on which tool was used.

5.3.2. Pairwise overlap coefficients

In order to better explore the intersections of assignments between software, we looked at pairwise Szymkiewicz–Simpson overlap coefficients. Defined as the proportion of the smaller set that is shared with the larger:

$$Overlap(X,Y) = \frac{|X \bigcap Y|}{min(|X|,|Y|)}$$

This gives us a measure of the similarity between two tools, in their assignment of reads as viral, that controls for differences in set size. Results for pairwise overlap coefficients, as well as total assignments for each tool, are summarised in Fig. 5.2.

There was a great amount of variation in the interactions between tools in the datasets, but there were also some common trends. The Pea coinfection dataset showed a high amount of agreement between all of the tools (Fig. 5.2a), with a maximum 2.5-fold difference in the number of reads assigned, and large overlap coefficients. This reinforces the observations seen in figure 5.1 of a subset of reads that are unmistakably viral, regardless of the methodology used in their detection. The CaLiber Hogweed dataset (Fig. 5.2b) had a reduced amount of overlap compared to the Pea coinfection dataset, with Mash Screen showing a distinctly low overlap with

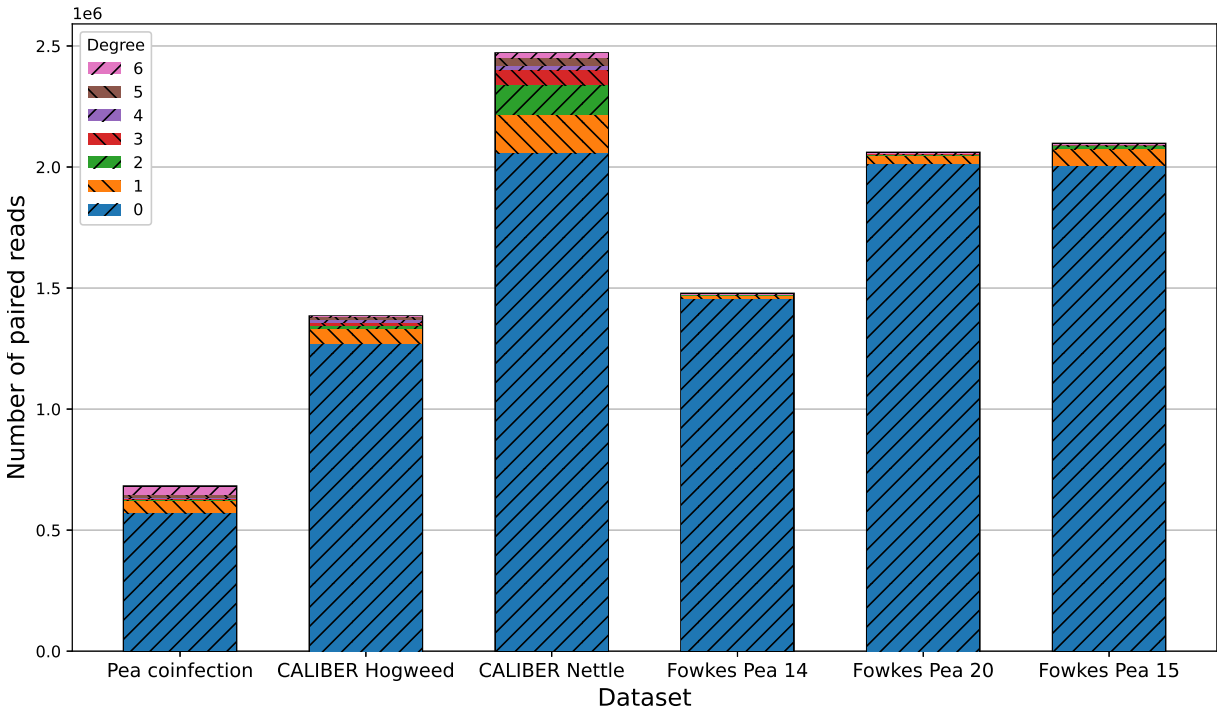
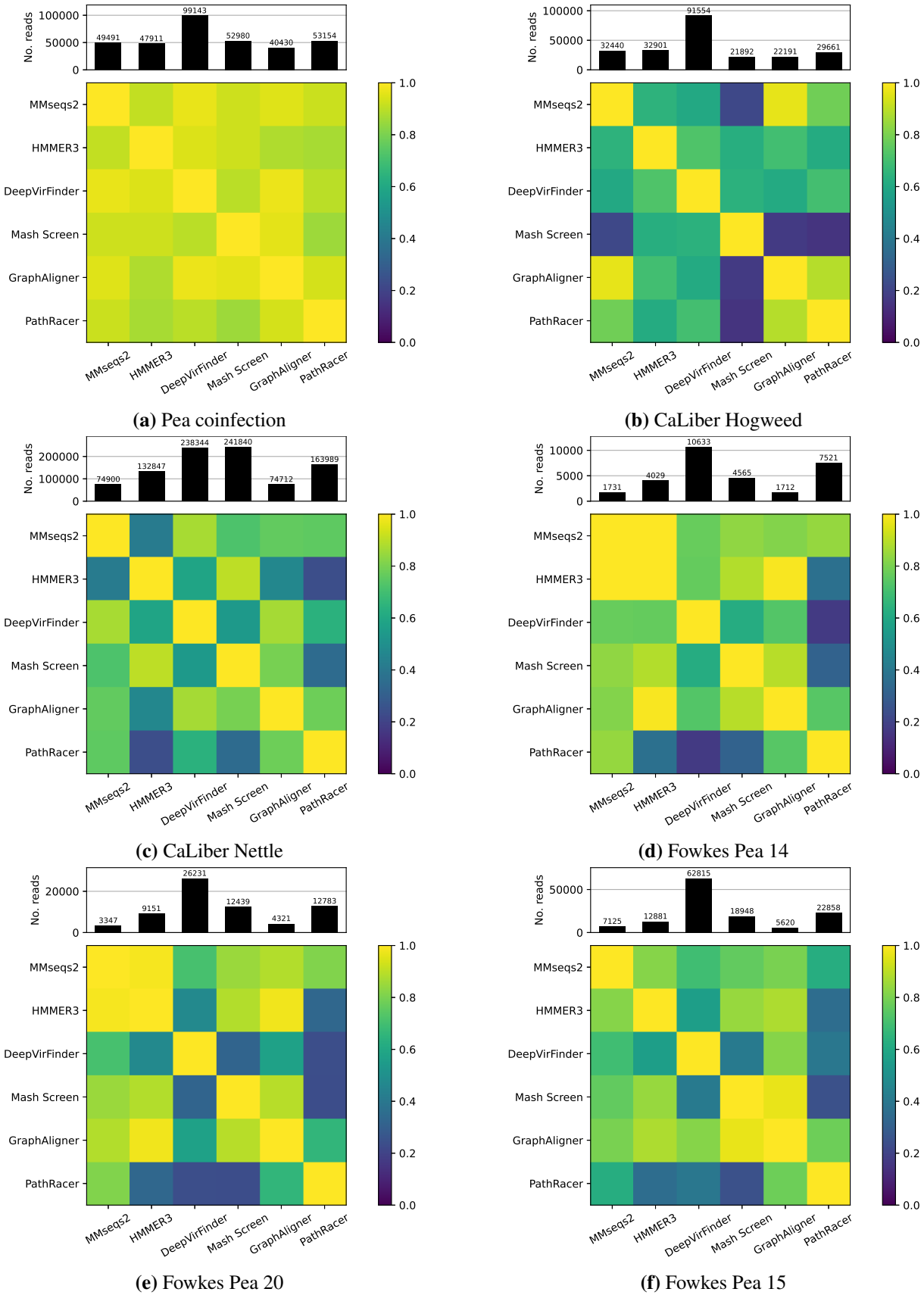


Figure 5.1 Number of reads in each dataset, partitioned by the degree of overlap of virus detection software assignments, i.e. the number of tools that agree that a read is of viral origin. A degree of zero, indicated by the bottom hatched area, signifies the fraction of reads that were not above viral detection threshold for any of the tested tools. These may represent reads originating from the host plant, endosymbionts, non-viral infections, organisms that have had direct contact with the samples

MMseqs2, GraphAligner, and PathRacer. GraphAligner, on the other hand, still showed a high overlap coeffient with PathRacer and MMseqs2, but there was a lower overlap between the latter two. This represents a more complex set of interactions between tools than simple clustering. Indeed, the mode of operation of GraphAligner has similarities to both other tools, using direct sequence alignment like MMseqs2, but operating on assembly graphs like PathRacer.

There was a larger difference in the total numbers of reads above detection thresholds, with DeepVirFinder reporting approximately 4.2 times more viral reads than mash screen. Other tools were more consistent in their relative proportions. The CaLiber Nettle dataset (Fig. 5.2c) once again showed a very different pattern of interaction. DeepVirFinder, MMseqs2, and GraphAligner had a high amount of overlap, with the latter two showing a diminished association. HMMER3 and Mash Screen also showed a high overlap with each other, but not with the previously mentioned set (DeepVirFinder, MMseqs2, and GraphAligner). This clustering supports the degree two and three overlaps seen in Figure 5.1. Relative reported viral read numbers were again high in DeepVirFinder, but also with Mash Screen and PathRacer. The clustering seen here does not seem to align with the modes of operation of the tools, where DeepVirFinder uses machine learning on contigs, MMseqs2 uses direct alignment on contigs, and GraphAligner uses direct alignment on assembly graphs. HMMER3 and Mash Screen are even more distinct, applying sequence models to contigs and analysing k-mer statistics from reads, respectively. While there does not seem to be a simple explanation to this clustering, the presence of an underlying pattern cannot be discounted.



(a) Pea coinfection

(b) CaLiber Hogweed

(c) CaLiber Nettle

(d) Fowkes Pea 14

(e) Fowkes Pea 20

(f) Fowkes Pea 15

Figure 5.2 Pairwise overlap coefficient of reads above viral detection threshold, as well as the total number of putative viral reads for each tool.

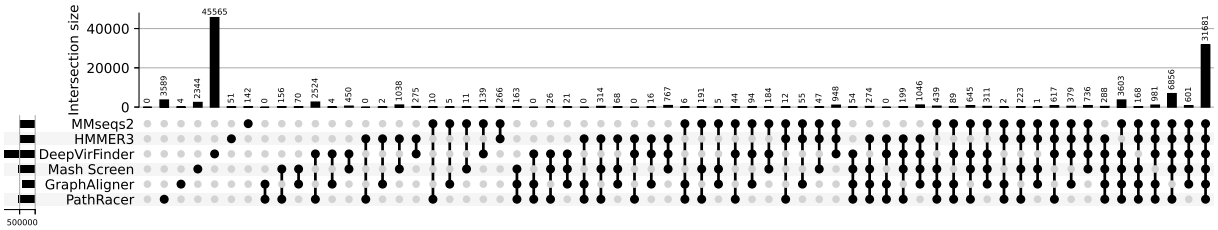
The three Fowkes Pea datasets Fig. 5.2d-5.2f had similar modes of interaction, which were once again distinct from the previous datasets. There was a clear cluster formed by MMseqs2, HMMER3, Mash Screen, and GraphAligner, with DeepVirFinder having a medium association with the cluster, while PathRacer only had significant overlap with GraphAligner. The pattern of reported viral read numbers was also consistent across these datasets, with DeepVirFinder reporting the greatest numbers, MMseqs2 and GraphAligner reporting the fewest, and Mash Screen and PathRacer reporting an intermediate amount. The consistency of outputs on these datasets is well explained, as datasets were generated from the same study (Fowkes et al., 2021), showing that the patterns of output of viral detection software depend on the underlying samples.

5.3.3. Higher level interactions between tools

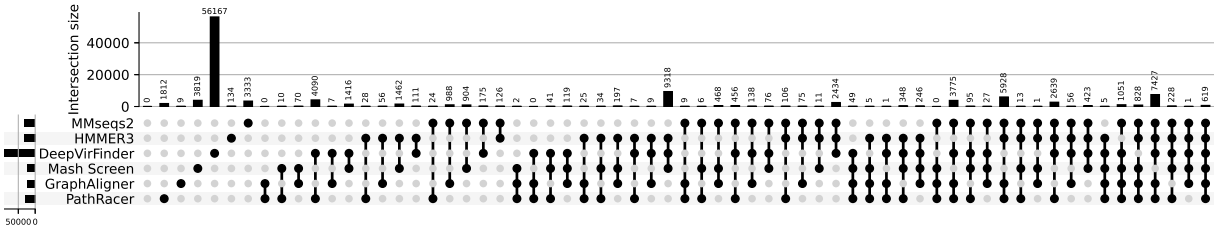
While pairwise overlap coefficients are effective in showing how much smaller sets are subsumed by larger ones, it does not give a good indication of the number of reads residing solely in the larger set, nor the higher level interactions between groups of virus detection software. To study the underlying patterns of interaction further, reads were binned into exclusive subsets, shown in Figure 5.3. Each subset was defined by a group of tools, and was composed of the reads that all defining tools had assigned as viral, with no additional tool labelling those reads as viral.

The unambiguously viral subset of the Pea coinfection dataset showed as a clear peak of unanimous assignment (Fig. 5.3a), making up approximately 4.75% of all reads, consistent with analysis of Figure 5.1 and 5.2a. All datasets displayed a substantial fraction labelled as viral by DeepVirFinder alone, in the CaLiber Hogweed dataset (Fig. 5.3b) this was to the extent of overwhelming all other subsets. DeepVirFinder only fractions explain the majority of degree one overlaps observed in Figure 5.1. The MMseqs2, GraphAligner, PathRacer (MGP) cluster inferred for the CaLiber Hogweed dataset in Figure 5.2b did not appear as a single peak in Figure 5.3b, but did feature in many relatively large peaks (MGP+DeepVirFinder, MGP+HMMER3, MGP+DeepVirFinder+HMMER3), accounting for the large overlap coefficients. Many datasets also featured a HMMER3, DeepVirFinder, and Mash Screen peak, as well as a HMMER3 and Mash Screen peak, and a DeepVirFinder and PathRacer peak. These were largely not seen in Figure 5.2, outside some HMMER3 and Mash Screen interactions in Figures 5.2c-5.2f.

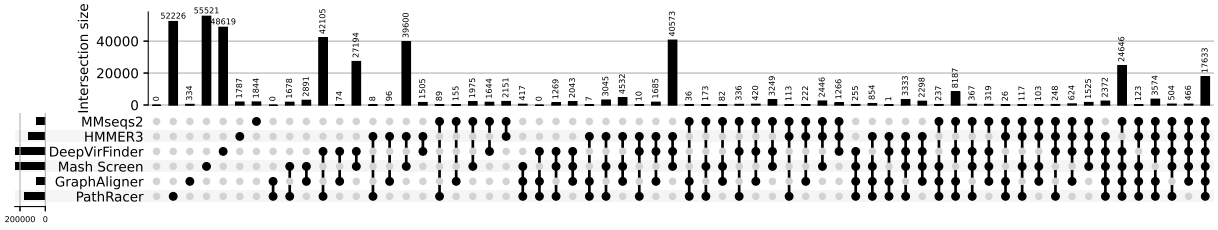
The complex patterns of overlap seen in Figure 5.2c were further explored in Figure 5.3c. This dataset (CaLiber Nettle) contained the most significant non-DeepVirFinder sole fractions, with PathRacer and Mash Screen also showing many uncorroborated viral assignments. Many other peaks involved HMMER3, DeepVirFinder, and Mash Screen in some combination, showing a tendency to label the same reads as being of viral origin. A unanimous fraction was visibly present in this dataset as well. The three Fowkes Pea datasets had a similar pattern of peaks, with DeepVirFinder, Mash Screen, and PathRacer sole fractions, the HMMER3, DeepVirFinder, and Mash Screen cluster and a visible unanimous peak. This reinforces conclusions from Figure 5.2 that the patterns of interaction between tools stem from underlying similarities in the samples.



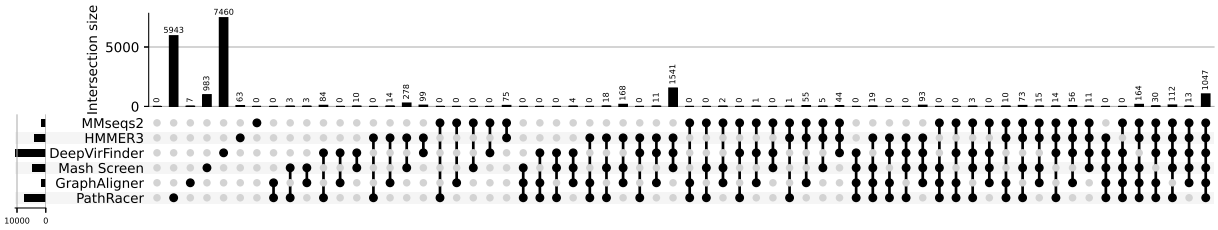
(a) Pea coinfection



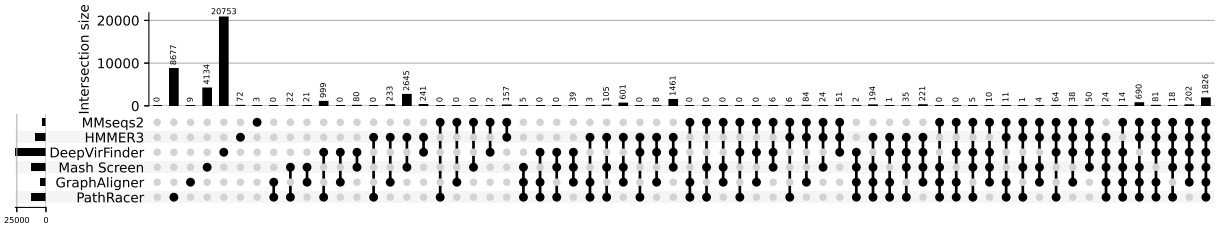
(b) CaLiber Hogweed



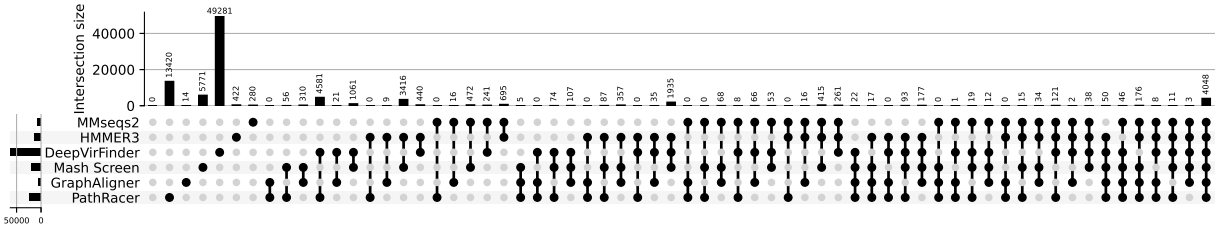
(c) CaLiber Nettle



(d) Fowkes Pea 14



(e) Fowkes Pea 20



(f) Fowkes Pea 15

Figure 5.3 UpSet plot of interactions between sets of viral assignments by viral detection software. Exclusive subsets are shown on the x-axis, with the tools that characterise that subset indicated with black circles connected by a line. The number of reads labelled by all tools in the subset, and no other tools, are plotted above the indicators. Reads with no viral assignments are not included. Additionally, the total number of assignments for each tool is plotted to the left of their names.

5.3.4. Analysis of read mappings

While there were stark differences in the labelling of reads as viral between tools, the biological relevance of this was not clear. To explore whether these software differ in the viral genomes they detect within datasets, we mapped all reads that were labelled as viral by at least one tool to the RefSeq viral genomes database. This mapping would only pick up a fraction of the labelled reads, so mappings were propagated to unmapped reads that co-assemble with those mapped. The target genomes and median alignment identities of the mappings could then be used to compare and contrast the viral hits of each tool. Additionally, hits that were corroborated by all tools were analysed as a 'unanimous' mapping.

Pea coinfection dataset

In the Pea coinfection dataset. The most common target viral genome for all tools was Pea enation mosaic virus-1, with 24,800-51,600 reads, representing 52.0% (DeepVirFinder) to 61.3% (GraphAligner) of all putative viral reads (Table 5.2). DeepVirFinder had the greatest number of labelled reads (51,600) mapping to this genome, and GraphAligner the least (24,800), with the rest showing similar numbers (28,742-31,989), though this was the case for almost all genomes. The unanimously labelled fraction for this genome was of size 19,844, this was smaller (80.0% the size) than the number of the smallest hit set (24,800), but still indicated a high overlap between all tools. The next most numerous genomes, in terms of total reads labelled as viral that mapped to them, were Pea enation mosaic virus-2 and Turnip yellows virus, with 5,408-8,770 and 3,406-5,754 reads respectively. There was a general similarity in the numbers of hits between tools in these and other plant-infecting viruses, with tools exhibiting a maximum of approximately a two-fold difference in read numbers. Plant-infecting viral genomes with greater than 100 reads also generally exhibited a high (>90%) median identity. These highly-mapped plant virus genomes provide a large part of the unanimous fraction seen in Figures 5.1 and 5.3a, as well as high overlap coefficients in 5.2b.

A high identity in the most mapped genomes was not observed for viruses with non-plant hosts, especially for bacteria-infecting viruses. Salmonella phage TS13 had many labelled reads mapped to it (258-4,848), especially by DeepVirFinder (4,848), but only 59 of these were labelled by all tools, and had a median of 21% identity in these mappings. As two random sequences, with equal Adenine:Uracil:Guanine:Cytosine ratios, would expect to have approximately 25% identity by chance alone, these hits must be approached with caution. This was similarly the case for other bacteriophages, such as Aeribacillus phage AP45 (105-1504 reads, 21 unanimous, at 14% median identity), Escherichia phage vB_EcoM_KAW1E185 (10-260, zero unanimous, at 16% identity), and Synechococcus phage S-B28 (5-129, zero unanimous, at 27% identity). Additionally, there were some low identity hits for insect viruses (Choristoneura fumiferana granulovirus with 21-339 reads, four unanimous, at 21-28% identity, and a host of the Eastern Spruce Budworm *Choristoneura fumiferana*), and mammalian viruses (Bubaline alphaherpesvirus 1 strain b6; 14-211 reads, 7 unanimous, at 18-29% identity; A host

of the Water Buffalo *Bubalus bubalis*), including a human virus (Hepatitis C virus genotype 1; 3-47 reads, one unanimous, at 17-22% identity). The low identity of these hits, the very different numbers of reads labelled as viral by each tool, a small fraction unanimous, as well as the fact that many of the hosts are unlikely to be present in the UK, leads to the conclusion that these hits are likely to be false positives from non-viral reads. On the other hand, this low identity may indicate the presence of viral genomes so divergent that they show little identity to any genome in our reference database.

In between the two extremes of identity, are genomes that represent a distantly related virus to one that may be found in the samples that the dataset was generated from. Alfalfa enamovirus-1 isolate Manfred is one such viral genome. With a median identity of 51-52%, a plant host, and a consistent number of putative reads across tools (588-760, 479 unanimous), it is likely related to a present virus. Indeed, it is known to be related to Pea enation mosaic virus-1 (Lu et al., 2022), which had many hits in this dataset. A similar situation was observed for Beet chlorosis virus, which is in the same Genus as Turnip yellows virus.

There were many genomes with low numbers of hits, 45 of which only had a single read labelled as viral, all but one by DeepVirFinder. While genomes with few (<10) hits mostly showed a low identity, there were some exceptions, notably Klebsiella phage ST15-OXA48phi14.1 and Klebsiella phage ST437-OXA245phi4.1. With the former having five to six hits across tools at a median identity of 90%, and the latter having two hits across all tools with 92% median identity. As reads in this dataset originate from environmental samples, and the host of these strains, *Klebsiella pneumoniae*, being commonly found in soil (Bagley, 1985), combined with the consistency of assignment and high mapping identity, leads us to conclude that at least one Klebsiella phage genome was present in the originating sample of this dataset at low titre. Chickpea stunt disease associated virus (5-7 mapped reads at 68-69% identity), Pepper vein yellows virus (9-22 mapped reads at 48-51% identity), Hubei polero-like virus 1 strain WHCC118254 (3-4 mapped reads at 46% identity), and even some singly-mapped genomes (Botrytis virus F at 78% identity; Luffa aphid-borne yellows virus at 55% identity; Cowpea polerovirus 1 at 62% identity) also showed this pattern. The viruses that these genomes are similar to are either plant-infecting or infect insects that feed on plants (Mihara et al., 2016). Despite a low number of reads, and a somewhat distant homology, these hits carry a biological rationale, and may represent related genomes to those present in the original samples. DeepVirFinder, despite being highly prolific in assigning reads as viral, acts as a great indicator for putative viral genomes in this dataset when corroborated by a homology-based tool.

While most reads labelled as viral were able to be either mapped to, or co-assembled with reads that mapped to, a viral genome, some hits could not be associated with any genome. These reads were binned into a "Novel mapping" entry. There was a great difference in the number of such reads between tools, with the highest being those labelled by DeepVirFinder at 14,608 (14.7% of total reads labelled as viral by DeepVirFinder), the lowest by GraphAligner at 840 (2.0% of total), and the rest falling between 1,000-2,500 (MMseqs2 with 1,280; HMMER3 with 2,044; Mash Screen with 2,252; PathRacer with 2,085). The unanimously labelled fraction of these

reads (167 in total) was much smaller than that of any individual tool, indicating much disagreement between tools in this unmapped fraction. Whether these hits represent spurious results, or viral reads too distant, or of too low quality, to be mapped is unclear from these data.

Table 5.2 Viral genomes mapping to the Pea coinfection dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_003629.1 Pea enation mosaic virus-1	Count	30213	28742	51600	31761	24800	31989	19844
	Ident	95	95	95	95	95	95	95
NC_055479.1 Cabbage cytorhabdovirus 1 strain FER..	Count	773	703	943	786	662	793	547
	Ident	95	95	95	95	95	95	95
NC_004756.1 Beet western yellows virus	Count	259	236	310	261	230	268	185
	Ident	95	94	94	95	95	95	94
NC_055495.1 Faba bean polerovirus 1 strain 5253	Count	536	507	968	589	453	578	346
	Ident	93	93	92	93	93	93	93
NC_003743.1 Turnip yellows virus	Count	4135	3774	5754	4243	3406	4244	2744
	Ident	93	93	92	93	93	93	93
NC_016038.2 Brassica yellows virus isolate BrYV-..	Count	3652	3366	5365	3710	3024	3790	2425
	Ident	92	92	92	92	92	92	93
NC_049448.1 Klebsiella phage ST437-OXA245phi4.1	Count	2	2	2	2	2	2	2
	Ident	92	92	92	92	92	92	92
NC_003491.1 Beet mild yellowing virus	Count	356	318	377	359	300	356	249
	Ident	92	92	92	92	92	92	92
NC_003853.1 Pea enation mosaic virus-2	Count	6522	6060	8770	6679	5408	6746	4431
	Ident	91	91	91	91	91	91	91
NC_049454.1 Klebsiella phage ST15-OXA48phi14.1	Count	5	5	6	6	5	6	5
	Ident	90	90	90	90	90	90	90
NC_002766.1 Beet chlorosis virus	Count	96	85	114	98	76	97	62
	Ident	84	85	83	84	84	85	85
NC_002604.1 Botrytis virus F	Count	1	1	1	1	1	1	1
	Ident	78	78	78	78	78	78	78
NC_043419.1 Chickpea stunt disease associated vi..	Count	7	6	7	7	5	7	4
	Ident	69	68	69	69	69	69	75
NC_028112.1 Yellowstone lake phycodnavirus 1 DNA	Count	1	1	4	0	1	0	0
	Ident	67	67	47	0	67	0	0
NC_011544.1 Hosta virus X	Count	0	0	13	3	1	3	0
	Ident	0	0	51	59	64	41	0
NC_029302.1 Piscine myocarditis-like virus isola..	Count	1	3	20	3	1	2	0
	Ident	18	40	63	40	63	18	0
NC_034246.1 Cowpea polerovirus 1 isolate BE167	Count	1	1	1	1	0	1	0
	Ident	62	62	62	62	0	62	0
NC_001422.1 Escherichia phage phiX174	Count	0	1	28	1	0	1	0
	Ident	0	54	62	54	0	20	0
NC_032001.1 Only Syngen Nebraska virus 5	Count	0	0	7	0	0	1	0
	Ident	0	0	58	0	0	38	0
NC_033775.1 Noumeavirus isolate NMV1	Count	1	1	5	1	1	1	0
	Ident	56	56	55	56	56	56	0
NC_027703.1 Luffa aphid-borne yellows virus isol..	Count	1	1	1	1	1	1	1
	Ident	55	55	55	55	55	55	55
NC_029993.1 Alfalfa enamovirus-1 isolate Manfredi	Count	712	645	760	723	588	728	479
	Ident	52	51	51	52	52	52	52
NC_020864.1 Micromonas pusilla virus 12T genomic..	Count	1	2	10	2	2	3	1
	Ident	51	51	51	51	51	31	51
NC_015050.1 Pepper vein yellows virus genomic RNA	Count	7	5	6	8	6	7	4
	Ident	48	48	51	46	48	48	50
NC_036803.1 Pepper vein yellows virus 5 isolate ..	Count	2	0	2	1	2	2	0
	Ident	50	0	50	33	50	50	0
NC_025412.1 Melbournevirus isolate 1	Count	2	6	36	6	2	3	0
	Ident	25	25	25	49	26	25	0

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_008724.1 Acanthocystis turfacea Chlorella vir..	Count	1	1	4	0	0	0	0
	Ident	14	14	47	0	0	0	0
NC_055482.1 Pistachio ampelovirus A isolate W10	Count	0	0	1	0	0	0	0
	Ident	0	0	47	0	0	0	0
NC_034207.1 African eggplant yellowing virus iso..	Count	13	11	12	13	8	14	7
	Ident	47	47	40	47	40	40	47
NC_020486.1 Synechococcus phage S-RIM8 A.HR1	Count	5	9	35	9	5	4	1
	Ident	47	47	47	47	47	47	47
NC_015289.1 Synechococcus phage S-SSM5	Count	0	0	12	1	0	1	0
	Ident	0	0	46	23	0	40	0
NC_032224.1 Hubei polero-like virus 1 strain WHC..	Count	3	3	4	3	2	2	2
	Ident	46	46	46	46	46	46	46
NC_031032.1 Bacillus phage Stitch	Count	1	1	1	0	0	0	0
	Ident	46	46	46	0	0	0	0
NC_006658.1 Cotesia congregata virus complete ge..	Count	0	0	2	1	0	0	0
	Ident	0	0	45	25	0	0	0
NC_037056.1 Erysiphe necator mitovirus 3 isolate..	Count	0	0	3	0	0	0	0
	Ident	0	0	45	0	0	0	0
NC_047734.1 Cyanophage S-RIM44 isolate Np_42_0711	Count	0	0	1	0	0	1	0
	Ident	0	0	45	0	0	45	0
NC_015288.1 Prochlorococcus phage Syn1	Count	0	1	9	1	1	0	0
	Ident	0	45	36	45	45	0	0
NC_020855.1 Cyanophage P-RSM6 genomic sequence	Count	1	2	8	2	1	1	1
	Ident	39	44	39	44	39	39	39
NC_001479.1 Encephalomyocarditis virus	Count	1	2	19	2	0	2	0
	Ident	14	14	26	19	0	43	0
NC_029692.1 Brazilian marseillevirus strain BH2014	Count	1	1	8	0	0	0	0
	Ident	42	42	42	0	0	0	0
NC_055129.1 Pepper vein yellows virus 2 isolate Is	Count	9	11	22	12	9	14	7
	Ident	35	35	33	42	33	35	32
NC_021536.1 Synechococcus phage S-IOM18 genomic ..	Count	0	0	1	0	0	1	0
	Ident	0	0	41	0	0	41	0
NC_010732.1 Tobacco vein distorting virus	Count	7	7	7	7	6	7	6
	Ident	41	41	41	41	39	41	39
NC_009898.1 Paramecium bursaria Chlorella virus ..	Count	0	0	7	0	0	2	0
	Ident	0	0	41	0	0	32	0
NC_021484.1 Maize yellow dwarf virus-RMV	Count	7	7	9	9	6	9	4
	Ident	38	41	41	41	34	41	25
NC_030230.1 Tokyovirus A1 DNA	Count	0	0	4	0	0	0	0
	Ident	0	0	39	0	0	0	0
NC_009823.1 Hepatitis C virus genotype 2	Count	1	3	12	1	0	3	0
	Ident	39	39	32	19	0	33	0
NC_006560.1 Cercopithecine herpesvirus 2	Count	0	0	2	0	0	1	0
	Ident	0	0	39	0	0	39	0
NC_030225.1 Pepo aphid-borne yellows virus isola..	Count	1	1	1	1	1	1	1
	Ident	38	38	38	38	38	38	38
NC_014545.1 Cotton leafroll dwarf virus	Count	29	27	32	30	23	29	20
	Ident	38	38	37	38	38	38	36
NC_049342.1 Escherichia phage 500465-1	Count	0	0	1	0	0	0	0
	Ident	0	0	38	0	0	0	0
NC_029691.1 Ixeridium yellow mottle virus 1 isol..	Count	5	5	5	5	5	5	5
	Ident	38	38	38	38	38	38	38
NC_018874.1 Abalone herpesvirus Victoria/AUS/2009	Count	0	0	11	0	0	0	0
	Ident	0	0	38	0	0	0	0
NC_055139.1 Harp seal herpesvirus isolate FMV04-..	Count	1	1	7	1	0	1	0
	Ident	17	17	37	17	0	17	0
NC_032255.1 Plodia interpunctella granulovirus i..	Count	0	0	1	0	0	0	0
	Ident	0	0	37	0	0	0	0

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_048645.1 Cronobacter phage vB_CsaM_leB	Count	0	0	1	0	0	0	0
	Ident	0	0	36	0	0	0	0
NC_008586.1 Ecotropis obliqua NPV	Count	0	0	5	0	0	1	0
	Ident	0	0	36	0	0	27	0
NC_006820.1 Synechococcus phage S-PM2	Count	1	1	6	1	0	1	0
	Ident	27	27	35	34	0	25	0
NC_070962.1 Synechococcus phage S-SCSM1	Count	2	2	7	2	2	1	1
	Ident	34	27	28	27	27	28	28
NC_043223.1 Senegalvirus SSV-A contig6 genomic s..	Count	0	0	1	0	0	0	0
	Ident	0	0	34	0	0	0	0
NC_003624.1 Impatiens necrotic spot virus segmen..	Count	1	1	1	1	1	0	0
	Ident	34	34	34	34	34	0	0
NC_055564.1 Patrinia mild mottle virus	Count	1	0	0	1	0	1	0
	Ident	34	0	0	34	0	34	0
NC_034265.1 Tobacco virus 2	Count	3	6	28	6	3	3	1
	Ident	28	28	33	33	28	28	23
NC_020867.1 Synechococcus phage S-RIP1 genomic s..	Count	1	6	45	6	3	3	0
	Ident	32	22	28	23	32	32	0
NC_015467.1 Groundnut ringspot and Tomato chloro..	Count	0	0	1	0	0	0	0
	Ident	0	0	32	0	0	0	0
NC_020859.1 Synechococcus phage S-RIM2 R1_1999	Count	0	0	3	0	0	0	0
	Ident	0	0	32	0	0	0	0
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	6	8	40	4	3	6	0
	Ident	30	30	30	31	27	23	0
NC_006639.1 Cotesia congregata virus complete ge..	Count	0	0	3	0	0	2	0
	Ident	0	0	30	0	0	22	0
NC_002520.1 Amsacta moorei entomopoxvirus 'L'	Count	0	0	1	0	0	1	0
	Ident	0	0	30	0	0	30	0
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	1	1	9	2	1	2	1
	Ident	22	22	30	22	22	22	22
NC_010809.1 Melon aphid-borne yellows virus	Count	5	5	5	5	4	5	4
	Ident	30	30	30	30	30	30	30
NC_026924.1 Synechococcus phage ACG-2014g isolat..	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	7	11	53	12	5	11	0
	Ident	16	26	29	19	29	29	0
NC_048026.1 Synechococcus T7-like virus S-TIP37	Count	0	0	1	0	0	0	0
	Ident	0	0	29	0	0	0	0
NC_031922.1 Synechococcus phage S-CAM9 isolate 1..	Count	3	7	70	10	6	8	1
	Ident	29	27	27	27	27	27	27
NC_043307.1 Diolcogaster facetosa bracovirus seg..	Count	1	1	11	0	0	3	0
	Ident	11	11	29	0	0	29	0
NC_043054.1 Bubaline alphaherpesvirus 1 strain b6	Count	27	32	211	43	14	43	7
	Ident	18	19	29	19	18	19	17
NC_001747.1 Potato leafroll virus	Count	0	1	13	1	0	2	0
	Ident	0	21	24	21	0	29	0
NC_048049.1 Synechococcus phage S-T4	Count	6	6	76	8	2	11	1
	Ident	28	28	28	28	24	21	21
NC_022646.1 Clostera anastomosis granulovirus He..	Count	0	0	4	0	0	0	0
	Ident	0	0	28	0	0	0	0
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	16	24	105	22	11	12	3
	Ident	28	28	28	24	25	25	24
NC_008168.1 Choristoneura fumiferana granulovirus	Count	36	62	339	65	21	49	4
	Ident	21	24	22	24	28	26	22
NC_047838.1 Synechococcus phage Bellamy	Count	0	0	1	0	0	0	0
	Ident	0	0	28	0	0	0	0
NC_028962.1 Staphylococcus phage phiIPLA-C1C	Count	0	0	3	0	0	0	0
	Ident	0	0	28	0	0	0	0

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_013756.1 Marseillevirus marseillevirus strain..	Count	1	1	10	0	0	0	0
	Ident	27	27	26	0	0	0	0
NC_010671.1 Musca domestica salivary gland hyper..	Count	0	0	1	0	0	1	0
	Ident	0	0	27	0	0	27	0
NC_048171.1 Synechococcus phage S-B28	Count	14	22	129	16	5	14	0
	Ident	27	27	27	27	25	27	0
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	6	9	43	8	3	4	0
	Ident	27	27	26	24	26	23	0
NC_043329.1 Diolcogaster facetosa bracovirus seg..	Count	1	1	11	0	1	1	0
	Ident	22	22	27	0	22	22	0
NC_031747.1 White clover mottle virus genomic RNA	Count	1	1	1	1	1	1	1
	Ident	26	26	26	26	26	26	26
NC_019401.1 Cronobacter phage vB_CsaM_GAP32	Count	0	0	1	0	0	1	0
	Ident	0	0	14	0	0	26	0
NC_028663.1 Cyanophage P-TIM40	Count	9	10	79	4	3	11	1
	Ident	26	26	22	22	26	24	21
NC_071140.1 Escherichia phage ZCEC13	Count	0	1	0	1	0	0	0
	Ident	0	25	0	25	0	0	0
NC_031944.1 Synechococcus phage S-WAM1 isolate 0..	Count	5	12	60	13	5	7	3
	Ident	24	23	25	24	24	24	24
NC_028793.2 Phasey bean mild yellows virus isola..	Count	1	1	1	1	1	1	1
	Ident	25	25	25	25	25	25	25
NC_015283.1 Prochlorococcus phage P-RSM4	Count	1	1	3	1	1	2	0
	Ident	24	24	25	22	24	24	0
NC_021099.1 Hop trefoil cryptic virus 2 isolate ..	Count	0	2	19	6	0	2	0
	Ident	0	20	25	23	0	20	0
NC_013110.1 Primula malacoides virus China/Mar20..	Count	0	1	2	1	0	1	0
	Ident	0	20	24	20	0	17	0
NC_048015.1 Cyanophage S-TIM4	Count	1	2	5	1	1	0	0
	Ident	24	23	24	22	22	0	0
NC_026242.1 Tipula oleracea nudivirus isolate 35	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_014637.1 Cafeteria roenbergensis virus BV-PW1	Count	3	4	46	10	2	3	0
	Ident	20	20	14	23	16	23	0
NC_021072.1 Cyanophage Syn30 genomic sequence	Count	1	1	1	0	0	0	0
	Ident	23	23	23	0	0	0	0
NC_034247.1 Cowpea polerovirus 2 isolate BE179	Count	3	3	3	3	3	3	3
	Ident	23	23	23	23	23	23	23
NC_007646.1 Ovine herpesvirus 2 strain BJ1035	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_019491.1 Cyprinid herpesvirus 1 strain NG-J1	Count	0	2	11	2	1	2	0
	Ident	0	12	23	12	12	12	0
NC_028250.1 Rosellinia necatrix partitivirus 6 C..	Count	0	1	0	1	0	0	0
	Ident	0	23	0	23	0	0	0
NC_016657.1 Cyanophage 9515-10a	Count	4	7	54	10	2	7	0
	Ident	23	23	23	23	23	23	0
NC_004102.1 Hepatitis C virus genotype 1	Count	3	8	47	8	5	8	1
	Ident	17	17	18	17	17	22	17
NC_009827.1 Hepatitis C virus genotype 6	Count	2	4	23	5	2	1	0
	Ident	22	20	19	22	22	18	0
NC_008725.1 Maruca vitrata MNPV	Count	1	1	3	2	0	0	0
	Ident	22	22	12	19	0	0	0
NC_033436.1 Wuchan romanomermis nematode virus 2..	Count	0	0	3	0	0	1	0
	Ident	0	0	22	0	0	22	0
NC_021095.1 White clover cryptic virus 2 isolate..	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_001822.1 Leek white stripe virus	Count	6	4	6	6	6	6	4
	Ident	22	22	22	22	22	22	22

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_024697.1 Aureococcus anophagefferens virus is..	Count	1	1	11	1	1	4	0
	Ident	18	18	19	19	18	21	0
NC_038828.1 Heterobasidion RNA virus 1 isolate H..	Count	0	0	1	0	0	0	0
	Ident	0	0	21	0	0	0	0
NC_053004.1 Salmonella phage TS13	Count	425	669	4848	713	258	715	59
	Ident	21	21	21	21	21	21	21
NC_043508.1 Persea americana chrysovirus segment..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_043574.1 Cachoeira Porteira virus strain BeAr..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_020235.1 Rosellinia necatrix partitivirus 2 C..	Count	1	1	1	0	0	0	0
	Ident	20	20	15	0	0	0	0
NC_020875.1 Cyanophage S-SSM4 genomic sequence	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	20	0
NC_030379.1 Tospovirus kiwifruit/YXW/2014 segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_006882.2 Prochlorococcus phage P-SSP7	Count	1	1	13	1	1	2	0
	Ident	20	20	20	19	20	20	0
NC_004452.3 Beet black scorch virus	Count	1	1	1	1	1	1	1
	Ident	20	20	20	20	20	20	20
NC_038882.1 Hepatitis C virus strain H77 pCV-H77..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_033778.1 Leptopilina boulardi filamentous vir..	Count	1	0	4	1	0	0	0
	Ident	19	0	16	19	0	0	0
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	1	4	51	5	1	6	0
	Ident	16	19	19	19	19	16	0
NC_021312.1 Phaeocystis globosa virus strain 16T	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_022615.1 Dill cryptic virus 1 isolate IPP_hor..	Count	0	0	2	0	0	0	0
	Ident	0	0	18	0	0	0	0
NC_005902.1 Lymphocystis disease virus - isolate..	Count	0	0	1	0	0	0	0
	Ident	0	0	18	0	0	0	0
NC_071044.1 Bacillus phage vB_BanS_Nate	Count	0	0	1	0	0	0	0
	Ident	0	0	18	0	0	0	0
NC_013015.1 Sclerotinia sclerotiorum partitiviru..	Count	0	1	1	1	1	0	0
	Ident	0	18	18	18	18	0	0
NC_061448.1 Erwinia phage pEa_SNUABM_1	Count	0	0	1	0	0	0	0
	Ident	0	0	18	0	0	0	0
NC_004812.1 Macacine herpesvirus 1	Count	0	1	7	1	0	1	0
	Ident	0	18	18	18	0	18	0
NC_001632.1 Rice tungro spherical virus	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_030925.1 Bacillus phage Shbh1	Count	1	1	1	1	0	0	0
	Ident	17	17	17	17	0	0	0
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	18	36	260	36	10	38	0
	Ident	16	16	16	15	17	14	0
NC_044937.1 Paramecium bursaria Chlorella virus ..	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_047813.1 Staphylococcus phage Andhra	Count	7	9	54	7	4	4	1
	Ident	16	16	16	16	16	16	17
NC_070664.1 Streptococcus phage CHPC1062	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_043331.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_002816.1 Cydia pomonella granulovirus	Count	0	0	2	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_001782.1 Saccharomyces cerevisiae killer viru..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_037052.1 Pepper enamovirus isolate R1 ORF0	Count	3	3	3	3	3	3	3
	Ident	16	16	16	16	16	16	16
NC_052978.1 Proteus phage Saba	Count	6	11	65	7	6	8	1
	Ident	16	15	13	14	14	13	13
NC_020845.1 Cyanophage MED4-213	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_055365.1 Gordil virus isolate Dak ANBr 496d s..	Count	2	2	10	1	2	1	0
	Ident	16	16	16	16	16	16	0
NC_047992.1 Microbacterium phage Zeta1847	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_036600.1 Rosellinia necatrix partitivirus 8 g..	Count	1	0	1	0	0	0	0
	Ident	15	0	15	0	0	0	0
NC_049942.1 Escherichia phage JLK-2012	Count	0	0	7	2	0	1	0
	Ident	0	0	15	15	0	15	0
NC_024502.1 Gentian ovary ring-spot virus genomi..	Count	0	0	1	0	0	1	0
	Ident	0	0	15	0	0	15	0
NC_061449.1 Erwinia phage pEa_SNUABM_17	Count	3	1	14	3	0	2	0
	Ident	15	15	15	15	0	15	0
NC_048651.1 Aeribacillus phage AP45	Count	135	209	1504	229	105	220	21
	Ident	14	14	14	14	14	14	15
NC_028251.1 Rosellinia necatrix partitivirus 6 R..	Count	0	0	1	0	0	0	0
	Ident	0	0	15	0	0	0	0
NC_007609.1 Dulcamara mottle virus	Count	2	1	7	1	1	1	1
	Ident	15	15	15	15	15	15	15
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	16	27	166	25	11	23	1
	Ident	13	13	13	13	13	13	14
NC_006659.1 Cotesia congregata virus complete ge..	Count	0	0	4	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_003038.1 Invertebrate iridescent virus 6	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_055142.1 Lymphocryptovirus Macaca/pfe-lcl-E3	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_047880.1 Erwinia phage vB_EamM_Y3	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_016072.1 Megavirus chiliensis	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	14	21	168	18	8	21	2
	Ident	13	13	13	13	13	14	12
NC_026440.1 Pandoravirus inopinatum isolate KlaHel	Count	0	0	1	0	0	1	0
	Ident	0	0	14	0	0	14	0
NC_038553.1 Heterosigma akashiwo virus 01 isolat..	Count	3	5	28	5	2	5	0
	Ident	14	14	14	14	14	13	0
NC_049442.1 Arthrobacter phage KBurrousTX	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_014325.1 Bidens mottle virus	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_047815.1 Erwinia phage vB_EamM_Yoloswag	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_001266.1 Rabbit fibroma virus	Count	0	0	2	0	0	1	0
	Ident	0	0	12	0	0	13	0
NC_049372.1 Roseobacter phage RD-1410W1-01	Count	0	0	4	1	0	0	0
	Ident	0	0	13	13	0	0	0
NC_007921.1 Agrotis segetum nucleopolyhedrovirus	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_038832.1 Heterobasidion partitivirus 13 strai..	Count	1	1	3	0	0	1	0
	Ident	13	13	13	0	0	13	0
NC_048875.1 Pantoea phage vB_PagM_SSEM1	Count	1	1	1	0	1	1	0
	Ident	13	13	13	0	13	13	0

Table 5.2 Viral genomes mapping to the Pea coinfection dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_043313.1 Diolcogaster facetosa bracovirus clo..	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0
NC_028491.1 Diatraea saccharalis granulovirus	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_037665.1 Pandoravirus macleodensis	Count	0	0	4	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_049464.1 Serratia phage Muldoon	Count	1	1	16	1	1	2	0
	Ident	12	12	12	12	12	12	0
NC_022617.1 Red clover cryptic virus 1 isolate I..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_029304.2 Cnaphalocrocis medinalis granuloviru..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_048798.1 Klebsiella phage Marfa	Count	6	13	135	13	7	20	1
	Ident	12	12	12	12	12	12	12
NC_028095.1 Torulaspora delbrueckii dsRNA Mbarr-..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	8	7	120	7	3	12	2
	Ident	12	11	11	11	11	11	11
NC_003094.2 Helicoverpa armigera NPV	Count	0	0	0	1	0	0	0
	Ident	0	0	0	12	0	0	0
NC_030953.1 Shigella phage SHFML-11	Count	0	0	5	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_061451.1 Erwinia phage pEa_SNUABM_7	Count	0	0	1	1	0	0	0
	Ident	0	0	12	12	0	0	0
NC_002512.2 Rat cytomegalovirus Maastricht	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_006657.1 Cotesia congregata virus complete ge..	Count	0	2	14	3	1	1	0
	Ident	0	11	11	11	11	11	0
NC_023021.1 Formica exsecta virus 1 isolate Fex1	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
NC_006656.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	1	0
	Ident	0	0	11	0	0	11	0
NC_038752.1 RNA4: NCP=non-capsid protein	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
NC_043314.1 Diolcogaster facetosa bracovirus clo..	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
Novel mapping*	Count	1280	2044	14608	2252	840	2085	167
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

CaLiber Hogweed dataset

The same process was carried out for the CaLiber Hogweed dataset, summarised in Table 5.3. This dataset had shown a much lower agreement between tools compared to the Pea coinfection dataset in 5.2b and 5.3b, instead exhibiting mainly singular and clustered fractions. This disagreement was also seen in Table 5.3, where few genomes had a significant unanimous subset. Genomes that did show some unanimously assigned viral reads only had a small proportion of total assignments in this subset, such as Carrot betaflexivirus 2 isolate CBV-2 S15 with 597-1873 hits, of which just 13 being unanimous.

There were few genomes that showed an equal or greater than 80% median mapping identity, and none that showed an equal or greater than 90% identity. The two genomes which were above 80% were BeAn 58058 virus and Bellflower vein chlorosis virus isolate CT1. The former (BeAn 58058 virus) had a single read labelled by MMseqs2 at 81% identity, and no other tool. MMseqs2 is generally conservative to medium in the number of hits in its output (bar charts in figure 5.2), and this is the only genome in this dataset where MMseqs2 had a hit mapped, but no other tool did. BeAn 58058 virus is known to infect *Oryzomys sp.* rodents in the Amazon Region of Brazil (Wanzeller et al., 2017), so would be unlikely to be found in this plant sequencing dataset from the UK. At 81% median identity, this could indicate a more distantly related viral genome, but with just a single read from assembly-based software, this uncorroborated hit cannot be used to draw any conclusion. The latter virus with somewhat high identity, Bellflower vein chlorosis virus isolate CT1, showed a different pattern of hits. There were 35-144 reads mapping to it, none of which were unanimous. Mappings to this genome only had 80% median identity for HMMER3 labelled reads, with other tools showing a lower identity (66-77%), where MMseqs2 and GraphAligner showed the lowest. This variation in median identity was even more extreme in Brazilian marseillevirus strain BH2014, where DeepVirFinder showed a median 71% identity whereas the others only showed 33%, and the two unanimous reads only showing a 21% median identity. There was no clear pattern across this dataset with regard to median mapping identity across tools, with, for example, DeepVirFinder sometimes showing the highest identity (Brazilian marseillevirus strain BH2014) and sometimes the lowest (Yellowstone lake phycodnavirus 1).

The low amount of unanimity and differing mapping identities generally did not prevent the detection of at least one read by every tool of abundant genomes with medium to high identity. Indeed, there was still a strikingly similar number of hits on most plant virus genomes, aside from DeepVirFinder, even with few unanimous reads. This was the case for the previously mentioned Bellflower vein chlorosis virus isolate CT1 (34-52 hits excluding DeepVirFinder) and Carrot betaflexivirus 2 isolate CBV-2 S15 (412-655 hits excl. DeepVirFinder), as well as Carrot betaflexivirus 1 isolate CBV-1_S20 (374-652 hit excl. DeepVirFinder) and Black raspberry necrosis virus (457-695 hits excl. DeepVirFinder). This was also the case for some highly mapping, medium to high identity, non-plant viruses such as Synechococcus virus S-PRM1 (180-286 hits excl. DeepVirFinder).

Some highly mapping, low median identity, non-plant-infecting viral genomes were also observed in this dataset, similar to those seen in Table 5.2. This included the Aeribacillus phage AP45, Oxbow virus strain Ng1453, Choristoneura fumiferana granulovirus, and Hepatitis C virus genomes seen before, though in the latter case of genotype 6. Interestingly, there were some plant-infecting viral genomes that showed a similar pattern in this dataset, such as Pepper chlorotic spot virus isolate 14YV733 and Motherwort yellow mottle virus. Whether these genomes act as a catch-all for false positives, or represent genuine highly divergent genomes continues to be difficult to differentiate from these data.

This dataset also contained a large "Novel Mapping" fraction, similar to Table 5.2. Unlike the previous dataset, the number of these unmapped hits was generally consistent across tools, except for DeepVirFinder. GraphAligner showed the lowest as before, with 4996 reads, then Mash Screen with 5004, PathRacer with 6686, MMseqs2 with 7350, HMMER3 with 7441, and finally DeepVirFinder with 20659.

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_032111.1 BeAn 58058 virus	Count	1	0	0	0	0	0	0
	Ident	81	0	0	0	0	0	0
NC_027915.1 Bellflower vein chlorosis virus isol..	Count	47	52	144	35	34	46	0
	Ident	66	80	74	77	66	73	0
NC_025468.1 Carrot betaflexivirus 2 isolate CBV-..	Count	655	613	1873	412	451	597	13
	Ident	76	73	72	53	60	74	17
NC_029692.1 Brazilian marseillevirus strain BH2014	Count	96	87	241	58	58	90	2
	Ident	33	33	71	33	33	33	21
NC_020864.1 Micromonas pusilla virus 12T genomic..	Count	3	4	12	4	2	4	1
	Ident	67	57	57	47	67	57	38
NC_029302.1 Piscine myocarditis-like virus isola..	Count	38	35	96	22	25	32	0
	Ident	64	65	66	66	64	64	0
NC_020837.1 Synechococcus phage S-CAM1 genomic s..	Count	4	5	6	2	1	0	0
	Ident	66	66	66	66	66	0	0
NC_015326.1 Lausannevirus	Count	2	3	7	3	2	2	0
	Ident	62	62	45	27	62	62	0
NC_055365.1 Gordil virus isolate Dak ANBr 496d s..	Count	0	0	1	0	0	0	0
	Ident	0	0	62	0	0	0	0
NC_027925.1 Apis mellifera filamentous virus iso..	Count	6	5	19	5	4	4	1
	Ident	62	62	58	62	62	62	62
NC_040627.1 Lettuce chordovirus 1 isolate JG1	Count	1	1	2	0	0	0	0
	Ident	51	51	60	0	0	0	0
NC_025469.1 Carrot betaflexivirus 1 isolate CBV-..	Count	652	622	1684	374	439	571	7
	Ident	53	36	41	37	59	47	33
NC_034265.1 Tobacco virus 2	Count	2	1	6	0	1	1	0
	Ident	47	35	42	0	59	59	0
NC_001422.1 Escherichia phage phiX174	Count	20	23	41	11	14	17	0
	Ident	20	20	58	44	20	56	0
NC_025480.2 Carrot torradovirus 1	Count	657	633	1813	431	449	568	14
	Ident	41	25	35	57	41	41	14
NC_070960.1 Synechococcus phage S-H9-2	Count	1	5	19	5	1	3	0
	Ident	54	33	33	33	54	26	0
NC_028112.1 Yellowstone lake phycodnavirus 1 DNA	Count	19	18	51	12	17	21	0
	Ident	43	48	32	48	43	52	0
NC_004067.1 Pepino mosaic virus	Count	1	0	1	0	0	0	0
	Ident	20	0	52	0	0	0	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_055761.1 Synechococcus virus S-PRM1	Count	286	280	795	180	199	276	9
	Ident	41	52	36	41	52	48	20
NC_070961.1 Synechococcus phage S-H9-1	Count	3	1	1	0	1	1	0
	Ident	40	52	36	0	52	52	0
NC_013455.1 Sugarcane bacilliform Guadeloupe D v..	Count	0	0	1	0	0	0	0
	Ident	0	0	50	0	0	0	0
NC_043523.1 Angelica bushy stunt virus isolate AD	Count	1	1	2	0	1	1	0
	Ident	17	17	50	0	17	17	0
NC_003498.1 Carnation etched ring virus	Count	0	0	1	0	0	0	0
	Ident	0	0	50	0	0	0	0
NC_021071.1 Cyanophage P-RSM1 genomic sequence	Count	2	5	17	3	0	2	0
	Ident	45	38	38	35	0	48	0
NC_003056.1 Soybean dwarf virus genomic RNA	Count	7	7	13	3	4	4	0
	Ident	46	46	46	46	46	46	0
NC_055139.1 Harp seal herpesvirus isolate FMV04-..	Count	1	2	3	1	0	1	0
	Ident	20	33	20	45	0	20	0
NC_019516.1 Cyanophage S-TIM5	Count	0	2	2	2	0	1	0
	Ident	0	42	42	42	0	45	0
NC_031032.1 Bacillus phage Stitch	Count	5	5	19	4	4	6	0
	Ident	43	44	45	31	41	43	0
NC_019443.1 Synechococcus phage metaG-MbCM1	Count	6	4	14	2	5	5	0
	Ident	45	44	44	41	44	45	0
NC_040549.1 Apple-associated luteovirus isolate ..	Count	23	26	81	23	19	27	0
	Ident	19	19	44	19	19	19	0
NC_048049.1 Synechococcus phage S-T4	Count	1	1	4	0	0	0	0
	Ident	44	44	32	0	0	0	0
NC_019491.1 Cyprinid herpesvirus 1 strain NG-J1	Count	1	3	11	4	2	2	0
	Ident	37	37	44	37	37	37	0
NC_018072.1 Bean necrotic mosaic virus segment M	Count	35	30	90	26	19	26	0
	Ident	43	43	43	43	43	43	0
NC_006549.1 Singapore grouper iridovirus	Count	6	8	12	4	4	3	0
	Ident	22	32	22	42	42	39	0
NC_040606.1 Malacosoma neustria nucleopolyhedrov..	Count	1	0	1	1	1	1	0
	Ident	42	0	42	42	42	42	0
NC_055230.1 Akhmeta virus isolate Akhmeta_2013-88	Count	0	0	1	0	0	0	0
	Ident	0	0	41	0	0	0	0
NC_070962.1 Synechococcus phage S-SCSM1	Count	8223	8356	23194	5531	5664	7528	153
	Ident	34	41	31	36	34	41	17
NC_001782.1 Saccharomyces cerevisiae killer viru..	Count	6	6	14	1	5	5	0
	Ident	29	29	41	35	23	23	0
NC_002687.1 Ectocarpus siliculosus virus 1	Count	1	1	4	1	1	1	0
	Ident	41	41	41	41	41	41	0
NC_025479.2 Carrot torradovirus 1	Count	5	4	17	3	4	5	0
	Ident	21	26	40	25	27	21	0
NC_013221.1 Phytophthora infestans RNA virus 1 R..	Count	1	1	2	0	1	1	0
	Ident	39	39	29	0	39	39	0
NC_048102.1 Synechoccus phage S-P4	Count	123	127	323	80	88	113	5
	Ident	17	17	39	17	17	17	17
NC_007346.1 Emiliania huxleyi virus 86	Count	1	1	1	0	0	1	0
	Ident	36	36	36	0	0	36	0
NC_021072.1 Cyanophage Syn30 genomic sequence	Count	2	3	4	3	1	1	0
	Ident	26	34	33	36	32	32	0
NC_016072.1 Megavirus chiliensis	Count	0	0	1	0	0	0	0
	Ident	0	0	36	0	0	0	0
NC_011285.1 Mycobacterium phage Troll4	Count	0	0	1	0	0	0	0
	Ident	0	0	35	0	0	0	0
NC_021099.1 Hop trefoil cryptic virus 2 isolate ..	Count	7	10	20	9	3	3	0
	Ident	34	34	25	34	34	34	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_038828.1 Heterobasidion RNA virus 1 isolate H..	Count	6	5	23	3	5	7	1
	Ident	34	34	34	34	34	34	34
NC_036594.1 Orpheovirus IHUMI-LCC2 genome assembly	Count	0	1	1	1	0	0	0
	Ident	0	34	34	34	0	0	0
NC_025412.1 Melbournevirus isolate 1	Count	0	1	1	1	0	0	0
	Ident	0	34	34	34	0	0	0
NC_027719.1 Chrysanthemum stem necrosis virus is..	Count	0	0	1	0	0	0	0
	Ident	0	0	33	0	0	0	0
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	541	607	1611	404	376	525	9
	Ident	33	33	33	33	33	31	17
NC_043054.1 Bubaline alphaherpesvirus 1 strain b6	Count	3	4	9	2	1	2	0
	Ident	33	30	27	33	33	33	0
NC_025456.1 Synechococcus phage S-CBP1	Count	2	3	7	3	1	1	0
	Ident	28	27	33	29	27	27	0
NC_004102.1 Hepatitis C virus genotype 1	Count	228	237	660	149	153	186	3
	Ident	30	30	32	32	30	31	17
NC_009823.1 Hepatitis C virus genotype 2	Count	221	226	633	161	147	189	3
	Ident	32	32	32	32	32	32	32
NC_006883.2 Prochlorococcus phage P-SSM2	Count	0	1	3	1	0	0	0
	Ident	0	32	32	32	0	0	0
NC_006151.1 Suid herpesvirus 1	Count	89	93	245	63	57	70	0
	Ident	20	20	32	20	20	32	0
NC_048171.1 Synechococcus phage S-B28	Count	1	1	0	0	0	0	0
	Ident	32	32	0	0	0	0	0
NC_043346.1 Glyptapanteles indiensis bracovirus ..	Count	21	17	34	10	14	18	1
	Ident	31	31	31	31	31	12	31
NC_028955.1 Prochlorococcus phage P-TIM68	Count	12	8	31	7	11	16	0
	Ident	19	19	30	19	19	19	0
NC_019516.2 Cyanophage S-TIM5	Count	1	1	0	0	1	1	0
	Ident	30	30	0	0	30	30	0
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	91	89	277	58	57	82	2
	Ident	28	30	30	30	30	29	30
NC_040589.1 Diatom colony associated ssRNA virus..	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_009758.1 Marine RNA virus JP-B	Count	0	0	2	1	0	0	0
	Ident	0	0	30	27	0	0	0
NC_015287.1 Synechococcus phage S-SSM7	Count	3	3	12	3	2	4	0
	Ident	29	29	29	29	29	29	0
NC_031922.1 Synechococcus phage S-CAM9 isolate 1..	Count	1	0	1	0	0	0	0
	Ident	20	0	29	0	0	0	0
NC_008182.1 Black raspberry necrosis virus RNA1	Count	695	740	2047	457	486	651	14
	Ident	29	29	29	22	29	29	24
NC_031903.1 Synechococcus phage S-CAM22 isolate ..	Count	0	1	1	1	0	0	0
	Ident	0	28	28	28	0	0	0
NC_043329.1 Diolcogaster facetosa bracovirus seg..	Count	14	20	54	13	11	14	0
	Ident	27	21	27	21	22	27	0
NC_008168.1 Choristoneura fumiferana granulovirus	Count	4887	4970	13795	3264	3308	4511	96
	Ident	26	26	24	23	24	24	14
NC_005892.1 Sulfolobus turreted icosahedral virus	Count	0	0	3	1	0	0	0
	Ident	0	0	26	18	0	0	0
NC_020875.1 Cyanophage S-SSM4 genomic sequence	Count	1	2	3	1	1	1	0
	Ident	26	26	26	25	26	26	0
NC_002593.1 Plutella xylostella granulovirus	Count	3	4	8	6	2	1	0
	Ident	25	26	25	26	25	25	0
NC_028663.1 Cyanophage P-TIM40	Count	90	93	296	70	58	85	3
	Ident	24	25	24	24	24	24	24
NC_037666.1 Pandoravirus neocaledonia	Count	22	27	61	18	14	14	0
	Ident	19	18	18	25	19	13	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_043500.1 Wuhan Millipede Virus 2 strain WHWG0..	Count	0	0	1	0	0	0	0
	Ident	0	0	25	0	0	0	0
NC_021095.1 White clover cryptic virus 2 isolate..	Count	1	0	0	0	0	0	0
	Ident	25	0	0	0	0	0	0
NC_001747.1 Potato leafroll virus	Count	0	0	1	0	0	0	0
	Ident	0	0	25	0	0	0	0
NC_006560.1 Cercopithecine herpesvirus 2	Count	84	86	266	57	58	86	1
	Ident	15	15	24	15	15	15	15
NC_038882.1 Hepatitis C virus strain H77 pCV-H77..	Count	3	3	14	4	3	4	0
	Ident	21	21	22	24	21	21	0
NC_009827.1 Hepatitis C virus genotype 6	Count	25	24	67	22	20	22	2
	Ident	24	23	24	23	24	24	16
NC_015289.1 Synechococcus phage S-SSM5	Count	0	0	1	0	0	0	0
	Ident	0	0	24	0	0	0	0
NC_008518.1 Trichoplusia ni ascovirus 2c	Count	0	1	1	1	0	0	0
	Ident	0	24	24	24	0	0	0
NC_070765.1 Mycobacterium phage Onyinye	Count	0	0	0	1	0	1	0
	Ident	0	0	0	24	0	24	0
NC_013756.1 Marseillevirus marseillevirus strain..	Count	0	1	1	1	0	0	0
	Ident	0	23	23	23	0	0	0
NC_017940.1 European sheatfish virus	Count	4	2	9	1	4	4	0
	Ident	23	23	16	23	23	23	0
NC_048183.1 Synechococcus phage S-CBP4 genomic s..	Count	2	0	12	1	2	3	0
	Ident	23	0	23	23	23	23	0
NC_021097.1 Red clover cryptic virus 2 isolate I..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_025464.1 Synechococcus phage S-CBP4	Count	0	0	2	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_003616.1 Impatiens necrotic spot virus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_007921.1 Agrotis segetum nucleopolyhedrovirus	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_021148.1 Dill cryptic virus 2 isolate IPP_hor..	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_010490.1 Tomato zonate spot virus segment M	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_004560.1 Oyster mushroom spherical virus	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_003988.1 Simian enterovirus A	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_006553.1 Avian sapelovirus	Count	0	0	1	1	0	0	0
	Ident	0	0	22	14	0	0	0
NC_040361.1 Planarian secretory cell nidovirus i..	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_030925.1 Bacillus phage Shbh1	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_023162.1 Carp picornavirus 1 isolate F37/06	Count	18	19	34	15	11	16	1
	Ident	19	19	22	19	19	19	19
NC_004812.1 Macacine herpesvirus 1	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_054662.1 Streptomyces phage Omar	Count	4	4	7	4	4	3	0
	Ident	22	22	22	22	22	22	0
NC_038752.1 RNA4: NCP=non-capsid protein	Count	0	0	1	0	0	1	0
	Ident	0	0	21	0	0	21	0
NC_024382.1 Alcelaphine herpesvirus 2 isolate to..	Count	0	0	1	0	0	0	0
	Ident	0	0	21	0	0	0	0
NC_009127.1 Cyprinid herpesvirus 3	Count	13	11	23	6	6	9	0
	Ident	21	19	21	21	20	20	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_009013.1 Tomato torrado virus RNA1	Count	2	1	11	3	1	3	0
	Ident	19	18	19	17	21	21	0
NC_016447.1 Aotine herpesvirus 1 strain S34E	Count	3	4	21	6	1	4	0
	Ident	14	20	21	17	14	14	0
NC_037667.1 Pandoravirus quercus	Count	11	14	26	8	7	11	0
	Ident	17	19	13	21	17	17	0
NC_036600.1 Rosellinia necatrix partitivirus 8 g..	Count	14	12	34	4	7	7	0
	Ident	20	20	14	20	20	20	0
NC_031290.1 Whenzhou Shrimp Virus 1 strain BJDX-..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	20	0	0	0
NC_021312.1 Phaeocystis globosa virus strain 16T	Count	318	334	950	239	230	319	4
	Ident	15	15	15	15	15	20	15
NC_013110.1 Primula malacoides virus China/Mar20..	Count	0	2	4	2	0	0	0
	Ident	0	20	18	20	0	0	0
NC_053004.1 Salmonella phage TS13	Count	2785	2808	7854	1869	1882	2521	57
	Ident	20	20	20	20	20	20	20
NC_008580.1 Rabbit vesivirus	Count	1	0	0	1	0	0	0
	Ident	20	0	0	18	0	0	0
NC_022098.1 Pandoravirus salinus	Count	51	55	139	46	37	45	1
	Ident	19	19	14	20	19	20	19
NC_001348.1 Human herpesvirus 3	Count	1	1	7	2	1	1	0
	Ident	20	15	18	15	20	20	0
NC_001493.2 Ictalurid herpesvirus 1 strain Aubur..	Count	18	17	62	9	12	18	0
	Ident	18	15	20	13	15	17	0
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	252	255	641	148	166	222	5
	Ident	17	18	18	18	17	18	20
NC_014765.1 Bathycoccus sp. RCC1105 virus BpV1	Count	2	1	7	0	1	2	0
	Ident	19	19	20	0	19	19	0
NC_043569.1 Iaco virus strain BeAn314206 nucleoc..	Count	9	8	37	6	7	9	1
	Ident	20	19	18	20	19	19	19
NC_018874.1 Abalone herpesvirus Victoria/AUS/2009	Count	6	8	25	6	5	7	0
	Ident	16	16	13	20	16	16	0
NC_002794.1 Tupaiid herpesvirus 1	Count	1	1	0	0	1	1	0
	Ident	20	20	0	0	20	20	0
NC_010356.1 Glossina pallidipes salivary gland h..	Count	2	1	1	1	2	1	0
	Ident	20	20	20	17	20	20	0
NC_030230.1 Tokyovirus A1 DNA	Count	5	6	13	7	2	3	0
	Ident	19	19	19	19	19	19	0
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	462	454	1280	310	296	432	9
	Ident	19	19	19	19	19	19	19
NC_021858.1 Pandoravirus dulcis	Count	16	15	42	8	10	15	0
	Ident	17	15	18	16	19	18	0
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	42	44	104	27	31	39	0
	Ident	19	19	19	19	19	19	0
NC_020231.1 Caviid herpesvirus 2 strain 21222	Count	19	18	41	5	14	17	0
	Ident	16	16	15	19	19	15	0
NC_005261.3 Bovine herpesvirus 5 strain SV507/99	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_008198.1 Mycobacterium phage PBI1	Count	6	9	41	7	6	10	0
	Ident	19	19	17	18	19	19	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	14	21	72	21	7	13	1
	Ident	18	19	18	19	19	16	19
NC_035117.1 Common bottlenose dolphin gammaherpe..	Count	1	1	1	0	1	1	0
	Ident	18	18	18	0	18	18	0
NC_048053.1 Dickeya phage vB_DsoM_JA29	Count	0	0	1	0	0	0	0
	Ident	0	0	18	0	0	0	0
NC_042013.1 Agrobacterium phage Atu_ph07	Count	0	0	1	0	0	0	0
	Ident	0	0	18	0	0	0	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_037665.1 Pandoravirus macleodensis	Count	52	51	175	42	36	46	2
	Ident	18	17	17	13	18	17	13
NC_043103.1 Bat astrovirus Tm/Guangxi/LD38/2007 ..	Count	0	1	1	1	0	0	0
	Ident	0	17	17	17	0	0	0
NC_001479.1 Encephalomyocarditis virus	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_006639.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_047813.1 Staphylococcus phage Andhra	Count	529	554	1509	370	367	490	14
	Ident	17	17	17	17	17	17	17
NC_042048.1 Listeria phage LMTA-34	Count	8	7	13	3	5	7	0
	Ident	17	17	17	17	17	17	0
NC_049919.1 Escherichia phage SH2026Stx1	Count	1	1	1	0	1	1	0
	Ident	17	17	17	0	17	17	0
NC_043314.1 Diolcogaster facetosa bracovirus clo..	Count	4	3	8	3	4	4	0
	Ident	17	17	17	17	17	17	0
NC_026618.2 Mulberry vein banding virus isolate ..	Count	5	4	4	3	4	4	1
	Ident	17	17	17	12	17	17	17
NC_003038.1 Invertebrate iridescent virus 6	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_002327.1 Rice grassy stunt virus RNA 5	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_003445.1 Strawberry mottle virus RNA1 gene fo..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_022615.1 Dill cryptic virus 1 isolate IPP_hor..	Count	1	1	2	1	1	1	0
	Ident	16	13	14	13	16	16	0
NC_022332.1 Eel picornavirus 1 strain F15/05	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_048072.1 Streptomyces phage Darolandstone	Count	3	4	14	3	1	1	0
	Ident	16	16	16	16	16	16	0
NC_028250.1 Rosellinia necatrix partitivirus 6 C..	Count	2	4	9	3	1	1	0
	Ident	16	16	16	16	16	16	0
NC_006882.2 Prochlorococcus phage P-SSP7	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	16	0
NC_038825.1 Flammulina velutipes browning virus ..	Count	5	1	8	4	3	3	0
	Ident	16	16	16	16	16	16	0
NC_001266.1 Rabbit fibroma virus	Count	1	0	3	2	0	0	0
	Ident	14	0	16	16	0	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	5	5	9	1	3	5	0
	Ident	15	15	14	16	16	14	0
NC_011421.1 Bacillus phage SPO1	Count	3	2	7	1	2	3	0
	Ident	15	15	14	16	15	15	0
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	646	687	1798	467	445	597	12
	Ident	15	15	15	15	15	15	16
NC_037660.1 Botrytis cinerea fusarivirus 1	Count	1	1	1	0	0	1	0
	Ident	15	15	15	0	0	15	0
NC_035201.1 Ailuropoda melanoleuca papillomaviru..	Count	3	4	9	2	4	3	0
	Ident	15	15	15	15	15	15	0
NC_049372.1 Roseobacter phage RD-1410W1-01	Count	1	1	0	0	1	1	0
	Ident	15	15	0	0	15	15	0
NC_015492.1 Grapevine Bulgarian latent virus seg..	Count	0	0	1	0	0	1	0
	Ident	0	0	15	0	0	15	0
NC_043313.1 Diolcogaster facetosa bracovirus clo..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	15	0	0	0
NC_019495.1 Cyprinid herpesvirus 2 strain ST-J1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	15	0	0	0
NC_043307.1 Diolcogaster facetosa bracovirus seg..	Count	1	1	6	0	1	2	0
	Ident	15	15	15	0	15	15	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_005068.1 Cryptophlebia leucotreta granulovirus	Count	6	7	13	6	5	7	0
	Ident	15	15	15	15	15	15	0
NC_024502.1 Gentian ovary ring-spot virus genomi..	Count	25	22	69	13	19	25	1
	Ident	15	15	15	15	15	15	15
NC_028478.1 Tomato brown rugose fruit virus isol..	Count	3	1	3	0	1	1	0
	Ident	15	15	15	0	15	15	0
NC_024697.1 Aureococcus anophagefferens virus is..	Count	22	22	68	19	16	19	1
	Ident	15	15	14	15	15	15	15
NC_035218.1 Motherwort yellow mottle virus	Count	917	927	2641	636	653	826	21
	Ident	13	13	13	13	13	13	15
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	0	0	3	0	0	0	0
	Ident	0	0	15	0	0	0	0
NC_048651.1 Aeribacillus phage AP45	Count	475	473	1318	293	343	434	6
	Ident	15	15	15	15	15	15	15
NC_033829.1 Kallithea virus isolate DrosEU46_Kha..	Count	1	1	1	1	0	0	0
	Ident	15	15	15	15	0	0	0
NC_008862.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_001672.1 Tick-borne encephalitis virus	Count	0	0	1	0	0	1	0
	Ident	0	0	14	0	0	14	0
NC_038320.1 Carrot necrotic dieback virus strain..	Count	20	20	63	12	17	17	0
	Ident	14	14	14	14	14	14	0
NC_055142.1 Lymphocryptovirus Macaca/pfe-lcl-E3	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_024709.1 Ball python nidovirus strain 07-53	Count	0	0	2	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_003626.1 Maize chlorotic dwarf virus	Count	1	1	1	0	1	1	0
	Ident	14	14	14	0	14	14	0
NC_043566.1 Anhembi virus strain SPAr2984 nucleo..	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_023021.1 Formica exsecta virus 1 isolate Fex1	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_043508.1 Persea americana chrysovirus segment..	Count	4	3	3	1	1	2	0
	Ident	14	14	14	14	14	14	0
NC_033780.2 Mythimna unipuncta granulovirus B is..	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_055639.1 Bern perch virus strain BEPV CH17 se..	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_006659.1 Cotesia congregata virus complete ge..	Count	2	2	3	1	2	2	0
	Ident	13	13	14	13	13	13	0
NC_008913.1 Glypta fumiferanae ichnovirus segmen..	Count	1	1	1	0	0	1	0
	Ident	13	13	13	0	0	13	0
NC_013015.1 Sclerotinia sclerotiorum partitiviru..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	13	0
NC_002816.1 Cydia pomonella granulovirus	Count	0	0	1	0	0	1	0
	Ident	0	0	13	0	0	13	0
NC_003537.1 Dasheen mosaic virus	Count	1	1	0	0	1	1	0
	Ident	13	13	0	0	13	13	0
NC_048798.1 Klebsiella phage Marfa	Count	1	1	1	0	1	1	0
	Ident	11	11	13	0	11	11	0
NC_003836.1 Tomato aspermy virus RNA 3	Count	1	0	0	1	1	0	0
	Ident	13	0	0	13	13	0	0
NC_028461.1 Epizootic haematopoietic necrosis vi..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	13	0	0	0
NC_026440.1 Pandoravirus inopinatum isolate KlaHel	Count	1	2	4	1	1	1	0
	Ident	11	12	13	13	11	11	0
NC_055577.1 Physalis rugose mosaic virus isolate..	Count	3	2	4	0	3	3	0
	Ident	13	13	13	0	13	13	0

Table 5.3 Viral genomes mapping to the CALIBER Hogweed dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_043138.1 Angelica virus Y isolate AnVY-g poly..	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_038933.1 Lychnis ringspot virus RNA for gamma..	Count	0	1	2	1	0	0	0
	Ident	0	11	12	11	0	0	0
NC_040401.1 Tea plant necrotic ring blotch virus..	Count	1	1	1	0	1	1	0
	Ident	12	12	12	0	12	12	0
NC_038832.1 Heterobasidion partitivirus 13 strai..	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0
NC_037664.1 Botrytis cinerea hypovirus 1 satelli..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_038553.1 Heterosigma akashiwo virus 01 isolat..	Count	1	1	1	1	0	0	0
	Ident	12	12	12	12	0	0	0
NC_049392.1 Escherichia phage ESSI2_ev239 genome..	Count	1	1	7	1	0	1	0
	Ident	12	12	12	12	0	12	0
NC_040699.1 Drosophila innubila nudivirus isolat..	Count	0	0	2	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_040536.1 Esparto virus isolate SRR3939042_Esp..	Count	1	0	1	0	1	1	0
	Ident	12	0	12	0	12	12	0
NC_014637.1 Cafeteria roenbergensis virus BV-PW1	Count	1	1	2	0	1	1	0
	Ident	12	12	12	0	12	12	0
NC_034241.1 Diabrotica virgifera virgifera virus..	Count	19	17	54	13	11	16	1
	Ident	12	12	12	12	12	12	12
NC_007242.1 Vicia cryptic virus RNA2	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0
NC_052978.1 Proteus phage Saba	Count	0	0	3	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	223	221	648	164	170	234	4
	Ident	11	11	11	11	11	11	11
NC_038957.1 Picornavirus HK21 polyprotein gene	Count	8	11	23	11	3	3	0
	Ident	11	11	11	11	11	11	0
NC_024114.1 Jingmen Tick Virus isolate SY84 segm..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	11	0	0	0
NC_043352.1 Glyptapanteles indiensis bracovirus ..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	11	0
NC_026769.1 Bat polyomavirus 6c DNA	Count	1	1	1	0	0	1	0
	Ident	11	11	11	0	0	11	0
NC_027867.1 Mollivirus sibericum isolate P1084-T	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
NC_006651.1 Cotesia congregata virus complete ge..	Count	0	0	2	0	0	0	0
	Ident	0	0	11	0	0	0	0
NC_037663.1 Botrytis cinerea hypovirus 1 satelli..	Count	0	1	0	0	0	0	0
	Ident	0	11	0	0	0	0	0
Novel mapping*	Count	7350	7441	20659	5004	4996	6686	137
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

CaLiber Nettle dataset

The CaLiber Nettle dataset had been seen to exhibit a generally fractured pattern of hits in Figure 5.3c, but with an otherwise large unanimous peak. This was explored further using viral genome read mappings for this dataset (Table 5.4). There was a notable number of plant-infecting viral genomes with a greater or equal to 80% median mapping identity and many (>1000) hits. Asparagus virus 2, a segmented, positive-strand RNA virus, had greater than 5000 hits across all tools for each of its genome segments, all of which had greater or equal to 86% identity. All segments also showed a great number of unanimous reads, with 1666-2896 per segment, contributing to the unanimous peak previously mentioned in Figure 5.3c. The tool with the highest number of hits mapping to Asparagus virus 2 RNA 1 and RNA 3 was DeepVirFinder, but the most prolific tool mapping to RNA 2 was Mash Screen. This increased rate of Mash Screen hits, compared to the previous datasets in Tables 5.2 and 5.3, was seen across all genomes in this dataset. Elm mottle virus and Citrus variegation virus, both in the same genus as Asparagus virus 2, *Ilarvirus*, also showed many mapped hits, with greater or equal to 83% median identity. Citrus variegation virus RNA 3 had an especially high number of hits (12780-43479), though with only 2743 unanimous. Elm mottle virus and Spinach latent virus, also in the same genus, had a somewhat higher divergence, at 67-73% identity, but otherwise showed a similar pattern of hits. These plant-infecting, closely related, highly mapping, minimally divergent genomes with a generally similar pattern of read numbers and identities across tools, are very likely to represent the presence of at least one *Ilarvirus* in the original sample that the dataset was generated from.

There was a different pattern of hits when it came to genomes outside *Ilarvirus*. Eptesicus fuscus gammaherpesvirus, for example, showed a more variable number of hits (21-110), a much wider range of median identites (17-83%), and few unanimous reads (5). This, combined with the knowledge that Eptesicus fuscus gammaherpesvirus resides mainly in bats in North America (Subudhi et al., 2018), makes us doubtful of the presence of a related genome in the dataset. Not all non-*Ilarvirus* genomes showed this same pattern. Escherichia phage phiX174 had numerous mapped reads (5702-22117), with a large number unanimous (837), and medium identity across tools (54-60%). Combined with the knowledge that this bacteriophage is known to infect *Escherichia coli*, which associates with plant rhizospheres (Habteselassie et al., 2010), leads us to conclude that a somewhat divergent bacteriophage genome likely exists within the original sample. This was also the case for other bacteriophages within this dataset, including Synechococcus phage ACG-2014g (as well as other Synechococcus phages), Prochlorococcus phage Syn1, and an uncultured phage, Uncultured phage_MedDCM-OCT-S45-C4, detected as part of a marine virome study (Mizuno et al., 2013). This dataset showed many unmapped ("Novel Mapping") reads, 8955-24547, and the largest unanimous non-mapping fraction seen in any of the datasets so far, at 1289.

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_032801.1 Changjiang crawfish virus 6 strain C..	Count	1	0	1	1	1	1	0
	Ident	92	0	92	92	92	92	0
NC_011809.1 Asparagus virus 2 RNA 2	Count	6755	8044	13863	13986	6748	10782	2896
	Ident	90	90	90	90	90	90	90
NC_011807.1 Asparagus virus 2 RNA 3	Count	5031	5309	9114	9093	5047	7260	2293
	Ident	88	87	86	86	88	87	89
NC_011808.1 Asparagus virus 2 RNA 1	Count	6579	10874	19538	19806	6480	13663	1666
	Ident	89	88	88	88	89	88	89
NC_003569.1 Elm mottle virus replicase gene	Count	2663	3967	6872	7009	2661	4946	930
	Ident	87	87	87	86	87	87	88
NC_003568.1 Elm mottle virus RNA 2	Count	608	873	1521	1523	592	1144	200
	Ident	86	84	82	82	86	83	88
NC_009537.1 Citrus variegation virus RNA1	Count	2086	1656	2819	2820	2115	2514	1097
	Ident	87	86	86	86	87	86	87
NC_009538.1 Citrus variegation virus RNA2	Count	1845	3647	6455	6581	1872	4363	393
	Ident	86	84	84	84	85	84	87
NC_009536.1 Citrus variegation virus RNA 3	Count	12832	23505	42479	42856	12780	28921	2743
	Ident	84	83	83	83	84	83	86
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	21	57	110	105	23	71	5
	Ident	30	83	53	60	17	27	17
NC_043329.1 Diolcogaster facetosa bracovirus seg..	Count	56	126	246	244	64	178	6
	Ident	56	47	27	74	80	32	19
NC_015284.1 Prochlorococcus phage P-HM2	Count	0	0	1	0	0	1	0
	Ident	0	0	78	0	0	78	0
NC_003546.1 Citrus leaf rugose virus RNA 3	Count	45	92	172	168	43	114	3
	Ident	75	74	75	75	75	74	73
NC_003808.1 Spinach latent virus putative replic..	Count	29	23	41	35	30	34	17
	Ident	67	73	70	70	69	69	61
NC_003809.1 Spinach latent virus putative polyme..	Count	32	21	38	40	33	36	19
	Ident	73	72	72	71	73	72	73
NC_003570.1 Elm mottle virus movement protein an..	Count	250	301	534	529	252	390	104
	Ident	68	73	72	72	68	70	70
NC_048026.1 Synechococcus T7-like virus S-TIP37	Count	0	1	1	1	0	1	0
	Ident	0	41	73	41	0	73	0
NC_015287.1 Synechococcus phage S-SSM7	Count	0	0	1	1	0	0	0
	Ident	0	0	70	28	0	0	0
NC_015282.1 Synechococcus phage S-SM1	Count	13	29	56	53	17	42	2
	Ident	68	67	68	68	66	68	56
NC_022646.1 Clostera anastomosis granulovirus He..	Count	4	5	6	9	5	7	2
	Ident	33	21	33	21	33	67	33
NC_007016.1 Macaca fuscata rhadinovirus	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	67	0
NC_039074.1 Tomato necrotic streak virus isolate..	Count	4	4	4	6	4	4	3
	Ident	66	46	66	60	66	66	64
NC_055744.1 Synechococcus phage S-B64	Count	8	8	23	21	10	12	1
	Ident	25	65	57	65	35	41	35
NC_029302.1 Piscine myocarditis-like virus isola..	Count	134	267	474	504	126	340	24
	Ident	65	56	63	62	64	64	58
NC_031900.1 Synechococcus phage S-CAM4 isolate 0..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	63	0	0	0
NC_003547.1 Citrus leaf rugose virus RNA 2	Count	16	11	21	20	16	20	8
	Ident	58	62	63	63	58	58	58
NC_006883.2 Prochlorococcus phage P-SSM2	Count	1	1	0	1	0	0	0
	Ident	46	46	0	60	0	0	0
NC_001422.1 Escherichia phage phiX174	Count	5719	12000	21457	22117	5702	14409	837
	Ident	60	58	54	58	59	59	60

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_055719.1 Synechococcus phage S-H35	Count	2	0	5	2	1	4	0
	Ident	44	0	57	45	32	60	0
NC_020859.1 Synechococcus phage S-RIM2 R1_1999	Count	24	34	60	63	18	48	2
	Ident	58	30	41	34	59	40	47
NC_026924.1 Synechococcus phage ACG-2014g isolat..	Count	1178	2443	4452	4479	1181	2991	177
	Ident	58	58	58	58	58	58	57
NC_033775.1 Noumeavirus isolate NMV1	Count	6	14	17	24	3	15	1
	Ident	58	44	50	44	52	51	46
NC_015288.1 Prochlorococcus phage Syn1	Count	2117	4495	7935	8189	2167	5306	355
	Ident	57	57	58	57	58	58	58
NC_047838.1 Synechococcus phage Bellamy	Count	2	2	7	4	2	2	1
	Ident	57	57	47	31	38	42	57
NC_062734.1 Synechococcus phage ACG-2014d isolat..	Count	2	5	6	8	4	3	1
	Ident	57	46	54	46	50	54	54
NC_047718.1 Synechococcus phage ACG-2014b isolat..	Count	1	1	2	2	0	0	0
	Ident	57	48	50	52	0	0	0
NC_031235.1 Cyanophage S-RIM32 isolate RW_108_0702	Count	1	1	1	1	1	1	1
	Ident	56	56	56	56	56	56	56
NC_047719.1 Synechococcus phage ACG-2014b isolat..	Count	0	2	2	2	1	1	0
	Ident	0	56	54	56	56	51	0
NC_040771.1 Tunis virus strain Brest/Ar/T2756 se..	Count	2	2	4	3	1	5	0
	Ident	16	14	14	12	56	34	0
NC_015569.1 Synechococcus phage S-CRM01	Count	1	1	2	2	1	2	0
	Ident	36	56	43	46	36	43	0
NC_061440.1 Erwinia phage pEa_SNUABM_22	Count	0	1	0	1	0	0	0
	Ident	0	56	0	56	0	0	0
NC_027130.1 Synechococcus phage ACG-2014b isolat..	Count	0	1	1	4	0	2	0
	Ident	0	56	56	45	0	55	0
NC_055710.1 Synechococcus phage S-CAM22 isolate ..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	56	0
NC_055139.1 Harp seal herpesvirus isolate FMV04-..	Count	0	1	0	1	0	0	0
	Ident	0	56	0	56	0	0	0
NC_070962.1 Synechococcus phage S-SCSM1	Count	2	1	3	3	3	2	0
	Ident	55	51	54	46	46	55	0
NC_047709.1 Uncultured phage_MedDCM-OCT-S45-C4 DNA	Count	128	263	485	471	139	323	19
	Ident	55	55	55	55	55	55	55
NC_023584.1 Synechococcus phage S-MbCM100	Count	880	1721	3141	3183	845	2084	130
	Ident	54	55	55	54	54	54	54
NC_047717.1 Cyanophage S-RIM12 isolate W1_08_0910	Count	0	0	1	1	0	0	0
	Ident	0	0	54	54	0	0	0
NC_048034.1 Uncultured phage MedDCM-OCT-S08-C41 ..	Count	1	0	3	0	1	2	0
	Ident	54	0	54	0	54	52	0
NC_031906.1 Synechococcus phage S-CAM3 isolate 1..	Count	0	3	1	4	0	0	0
	Ident	0	53	32	53	0	0	0
NC_003382.1 Citrus yellow mosaic virus	Count	0	0	0	1	0	0	0
	Ident	0	0	0	53	0	0	0
NC_029692.1 Brazilian marseillevirus strain BH2014	Count	11	10	24	25	8	19	1
	Ident	45	39	49	48	42	48	53
NC_020838.1 Synechococcus phage S-RIP2 genomic s..	Count	1	1	5	4	2	1	0
	Ident	53	37	46	53	53	53	0
NC_020851.1 Synechococcus phage S-SKS1 genomic s..	Count	1	3	5	5	1	4	0
	Ident	33	52	33	52	33	43	0
NC_021530.1 Synechococcus phage S-CAM8 strain S-..	Count	3	13	23	26	6	15	0
	Ident	48	43	52	52	52	52	0
NC_006882.2 Prochlorococcus phage P-SSP7	Count	2	1	2	3	1	2	1
	Ident	46	52	48	44	52	48	52
NC_039075.1 Tomato necrotic streak virus isolate..	Count	3	2	3	3	3	3	2
	Ident	52	52	52	52	52	52	52

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_032001.1 Only Syngen Nebraska virus 5	Count	21	32	72	75	18	51	1
	Ident	38	35	44	40	50	52	36
NC_031242.1 Cyanophage S-RIM50 isolate RW_29_0704	Count	4	8	9	10	4	8	0
	Ident	52	52	52	52	52	48	0
NC_023587.1 Synechococcus phage S-MbCM7	Count	1837	3849	7029	7145	1854	4704	253
	Ident	40	41	47	40	40	48	51
NC_032255.1 Plodia interpunctella granulovirus i..	Count	1	0	1	2	1	1	0
	Ident	51	0	51	38	51	51	0
NC_031465.1 Golden Marseillevirus	Count	14	29	47	46	15	32	3
	Ident	37	37	51	50	36	46	35
NC_031922.1 Synechococcus phage S-CAM9 isolate 1..	Count	0	0	1	0	0	0	0
	Ident	0	0	50	0	0	0	0
NC_015286.1 Synechococcus phage Syn19	Count	7	16	31	34	9	29	0
	Ident	40	50	42	42	44	46	0
NC_028112.1 Yellowstone lake phycodnavirus 1 DNA	Count	22	38	67	73	23	50	2
	Ident	48	33	37	36	50	45	48
NC_020837.1 Synechococcus phage S-CAM1 genomic s..	Count	15	29	65	60	12	36	2
	Ident	34	48	46	49	46	46	32
NC_013085.1 Synechococcus phage S-RSM4	Count	12	26	55	53	15	39	3
	Ident	46	35	48	23	23	26	46
NC_015289.1 Synechococcus phage S-SSM5	Count	81	152	300	262	87	209	10
	Ident	48	34	36	34	35	46	45
NC_025412.1 Melbournevirus isolate 1	Count	2	15	18	23	3	10	1
	Ident	39	24	27	24	30	26	48
NC_008724.1 Acanthocystis turfacea Chlorella vir..	Count	1	2	3	3	2	2	0
	Ident	17	40	25	48	32	23	0
NC_001716.2 Human herpesvirus 7	Count	0	1	0	1	0	0	0
	Ident	0	47	0	47	0	0	0
NC_006549.1 Singapore grouper iridovirus	Count	1	4	11	5	1	5	0
	Ident	46	43	43	41	40	43	0
NC_015521.1 Cutthroat trout virus	Count	1	8	11	11	0	5	0
	Ident	46	15	15	15	0	15	0
NC_001973.1 Lymantria dispar MNPV	Count	1	2	6	7	0	2	0
	Ident	27	46	37	39	0	37	0
NC_043500.1 Wuhan Millipede Virus 2 strain WHWG0..	Count	0	0	1	0	0	0	0
	Ident	0	0	46	0	0	0	0
NC_020875.1 Cyanophage S-SSM4 genomic sequence	Count	10	17	43	38	13	25	2
	Ident	45	37	43	45	44	43	45
NC_015279.1 Synechococcus phage S-SM2	Count	1	0	0	0	0	0	0
	Ident	45	0	0	0	0	0	0
NC_026923.1 Synechococcus phage ACG-2014d isolat..	Count	26	58	85	93	28	56	2
	Ident	44	42	43	38	45	43	29
NC_004812.1 Macacine herpesvirus 1	Count	1	0	3	5	2	2	0
	Ident	45	0	45	43	45	30	0
NC_048171.1 Synechococcus phage S-B28	Count	1104	2187	4069	4026	1089	2664	163
	Ident	44	44	44	44	44	44	43
NC_027132.1 Synechococcus phage ACG-2014i isolat..	Count	0	0	1	0	0	0	0
	Ident	0	0	44	0	0	0	0
NC_048015.1 Cyanophage S-TIM4	Count	788	1637	2975	3003	834	1966	131
	Ident	26	34	32	28	44	30	26
NC_019443.1 Synechococcus phage metaG-MbCM1	Count	4	6	11	14	4	7	0
	Ident	42	42	44	44	42	41	0
NC_028663.1 Cyanophage P-TIM40	Count	87	190	325	335	90	207	12
	Ident	43	42	43	43	43	43	42
NC_022615.1 Dill cryptic virus 1 isolate IPP_hor..	Count	14	32	60	55	17	44	3
	Ident	43	16	17	16	21	17	43
NC_006884.2 Prochlorococcus phage P-SSM4	Count	0	0	1	2	0	0	0
	Ident	0	0	17	42	0	0	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_028955.1 Prochlorococcus phage P-TIM68	Count	469	1031	1847	1934	484	1199	65
	Ident	42	42	42	42	41	42	31
NC_021072.1 Cyanophage Syn30 genomic sequence	Count	322	641	1193	1206	317	803	41
	Ident	38	42	40	42	30	41	28
NC_031032.1 Bacillus phage Stitch	Count	55	109	194	175	56	133	12
	Ident	32	41	35	37	34	37	20
NC_027925.1 Apis mellifera filamentous virus iso..	Count	0	1	2	1	0	2	0
	Ident	0	17	34	17	0	41	0
NC_003834.1 Tulare apple mosaic virus RNA2	Count	16	10	16	16	16	16	9
	Ident	39	40	39	40	39	39	40
NC_006560.1 Cercopithecine herpesvirus 2	Count	6	23	31	35	7	17	1
	Ident	40	32	32	27	40	40	20
NC_013221.1 Phytophthora infestans RNA virus 1 R..	Count	2	1	5	7	1	1	0
	Ident	31	40	37	39	28	28	0
NC_055761.1 Synechococcus virus S-PRM1	Count	0	0	1	2	0	1	0
	Ident	0	0	40	29	0	40	0
NC_006567.1 Fragaria chiloensis latent virus RNA 2	Count	3	2	7	8	1	3	0
	Ident	15	21	26	26	40	24	0
NC_002052.1 Tomato spotted wilt virus RNA L	Count	1	4	4	3	1	4	0
	Ident	20	40	20	40	20	40	0
NC_048102.1 Synechoccus phage S-P4	Count	14	28	51	53	19	30	1
	Ident	39	38	38	39	39	37	39
NC_015281.1 Synechococcus phage S-ShM2	Count	35	47	112	106	35	79	5
	Ident	39	36	37	34	39	37	17
NC_055299.1 Alstroemeria necrotic streak virus i..	Count	0	0	1	0	0	1	0
	Ident	0	0	20	0	0	39	0
NC_003548.1 Citrus leaf rugose virus RNA 1	Count	5	5	5	6	5	5	5
	Ident	39	39	39	39	39	39	39
NC_021536.1 Synechococcus phage S-IOM18 genomic ..	Count	0	1	2	2	0	2	0
	Ident	0	38	30	28	0	38	0
NC_047734.1 Cyanophage S-RIM44 isolate Np_42_0711	Count	1054	2206	3947	3947	1036	2558	134
	Ident	29	28	29	27	29	29	38
NC_001782.1 Saccharomyces cerevisiae killer viru..	Count	33	64	114	116	39	73	4
	Ident	35	24	29	29	35	37	21
NC_024382.1 Alcelaphine herpesvirus 2 isolate to..	Count	0	2	1	3	0	1	0
	Ident	0	26	37	26	0	37	0
NC_025456.1 Synechococcus phage S-CBP1	Count	765	1573	2857	2914	766	1882	117
	Ident	36	36	36	36	36	36	33
NC_021249.1 Choristoneura rosaceana entomopoxvir..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	36	0
NC_001479.1 Encephalomyocarditis virus	Count	26	53	72	81	22	46	3
	Ident	36	34	25	34	36	36	36
NC_040625.1 Rhynchobatus djiddensis adomavirus 1..	Count	0	1	1	1	0	0	0
	Ident	0	36	36	36	0	0	0
NC_005068.1 Cryptophlebia leucotreta granulovirus	Count	21	52	90	91	20	64	2
	Ident	17	17	17	35	17	17	17
NC_034265.1 Tobacco virus 2	Count	16	35	54	68	15	34	5
	Ident	35	35	35	35	35	35	35
NC_003739.1 Raspberry bushy dwarf virus RNA 1	Count	0	0	1	0	0	1	0
	Ident	0	0	35	0	0	35	0
NC_001747.1 Potato leafroll virus	Count	16	38	53	61	14	43	0
	Ident	35	35	35	35	24	35	0
NC_038828.1 Heterobasidion RNA virus 1 isolate H..	Count	43	85	130	145	35	97	5
	Ident	34	33	34	34	34	34	34
NC_070960.1 Synechococcus phage S-H9-2	Count	626	1339	2358	2431	619	1536	104
	Ident	34	34	34	34	34	34	34
NC_013953.1 Lymantria xylina MNPV	Count	5	8	13	13	5	9	3
	Ident	34	34	34	34	34	34	34

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_047992.1 Microbacterium phage Zeta1847	Count	0	1	1	1	0	0	0
	Ident	0	34	34	34	0	0	0
NC_015467.1 Groundnut ringspot and Tomato chloro..	Count	1	1	7	11	1	4	0
	Ident	33	33	33	34	33	29	0
NC_001664.4 Human betaherpesvirus 6A	Count	0	0	1	0	0	1	0
	Ident	0	0	33	0	0	33	0
NC_028095.1 Torulaspora delbrueckii dsRNA Mbarr-..	Count	5	9	12	16	3	14	0
	Ident	33	18	19	26	33	25	0
NC_031903.1 Synechococcus phage S-CAM22 isolate ..	Count	2	1	4	2	2	5	1
	Ident	31	32	31	31	31	32	32
NC_009823.1 Hepatitis C virus genotype 2	Count	380	747	1402	1378	364	945	51
	Ident	32	32	32	32	32	32	32
NC_070761.1 Gordonia phage GMA2	Count	1	6	8	6	1	4	0
	Ident	14	32	32	32	14	14	0
NC_001962.1 Bombyx mori NPV	Count	5	5	19	11	6	14	1
	Ident	17	31	31	31	17	17	17
NC_013756.1 Marseillevirus marseillevirus strain..	Count	20	46	78	76	22	46	4
	Ident	29	29	31	31	29	27	27
NC_003833.1 Tulare apple mosaic virus RNA1	Count	13	8	15	13	14	15	7
	Ident	31	26	31	31	31	31	31
NC_010489.1 Tomato zonate spot virus segment S	Count	5	9	17	20	3	12	0
	Ident	31	13	13	13	31	31	0
NC_018464.1 Shamonda virus N and NSs genes	Count	2	3	5	3	2	4	1
	Ident	29	24	28	24	29	29	31
NC_008361.1 Spodoptera frugiperda ascovirus 1a c..	Count	1	2	3	3	1	2	0
	Ident	31	31	31	31	31	31	0
NC_003810.1 Spinach latent virus putative moveme..	Count	5	6	14	9	5	9	1
	Ident	30	26	26	24	30	24	30
NC_003414.1 Ageratum yellow vein Singapore alpha..	Count	0	0	1	1	0	0	0
	Ident	0	0	30	30	0	0	0
NC_009758.1 Marine RNA virus JP-B	Count	1	1	0	1	1	0	0
	Ident	30	30	0	21	30	0	0
NC_005849.1 Parietaria mottle virus RNA 2	Count	1	1	2	2	1	2	1
	Ident	26	26	23	30	26	23	26
NC_003102.1 Spodoptera litura NPV	Count	4	5	12	10	2	7	0
	Ident	25	30	25	30	25	25	0
NC_031290.1 Whenzhou Shrimp Virus 1 strain BJDX-..	Count	1	1	0	1	0	0	0
	Ident	30	30	0	15	0	0	0
NC_020486.1 Synechococcus phage S-RIM8 A.HR1	Count	22	38	68	80	25	42	4
	Ident	28	21	24	26	30	24	20
NC_038386.1 Rhinolophus ferrumequinum circovirus 1	Count	0	0	1	1	0	0	0
	Ident	0	0	30	30	0	0	0
NC_003740.1 Raspberry bushy dwarf virus RNA 2	Count	0	0	2	0	0	2	0
	Ident	0	0	30	0	0	30	0
NC_030230.1 Tokyovirus A1 DNA	Count	0	4	4	5	0	3	0
	Ident	0	30	30	30	0	20	0
NC_026242.1 Tipula oleracea nudivirus isolate 35	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	30	0
NC_016659.1 Cyanophage NATL2A-133	Count	0	0	0	1	0	0	0
	Ident	0	0	0	30	0	0	0
NC_035117.1 Common bottlenose dolphin gammaherpe..	Count	6	15	32	24	5	27	1
	Ident	25	28	29	22	22	25	25
NC_021071.1 Cyanophage P-RSM1 genomic sequence	Count	0	1	0	1	0	0	0
	Ident	0	28	0	28	0	0	0
NC_015283.1 Prochlorococcus phage P-RSM4	Count	74	153	299	286	70	204	16
	Ident	28	28	28	28	28	28	28
NC_043054.1 Bubaline alphaherpesvirus 1 strain b6	Count	108	247	456	464	116	285	14
	Ident	20	19	21	28	21	20	23

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_048049.1 Synechococcus phage S-T4	Count	0	1	2	2	0	1	0
	Ident	0	28	27	27	0	20	0
NC_031935.1 Synechococcus phage S-WAM2 isolate 0..	Count	0	0	1	3	1	1	0
	Ident	0	0	28	28	28	28	0
NC_008580.1 Rabbit vesivirus	Count	7	15	35	31	6	29	1
	Ident	18	18	28	18	18	26	18
NC_004102.1 Hepatitis C virus genotype 1	Count	396	865	1586	1601	402	1035	49
	Ident	25	26	27	28	25	27	21
NC_039076.1 Tomato necrotic streak virus isolate..	Count	10	9	10	10	10	10	9
	Ident	27	27	27	27	27	27	27
NC_013801.1 Croton yellow vein mosaic alphasatel..	Count	1	0	1	0	0	0	0
	Ident	27	0	27	0	0	0	0
NC_005030.2 Tobacco leaf curl Yunnan virus satel..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	27	0
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	86	200	369	358	93	218	9
	Ident	25	24	25	25	25	26	27
NC_055482.1 Pistachio ampelovirus A isolate W10	Count	0	0	1	0	0	0	0
	Ident	0	0	27	0	0	0	0
NC_040743.1 Alstroemeria yellow spot virus isola..	Count	1	0	1	1	1	1	0
	Ident	26	0	26	26	26	26	0
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	570	1175	2017	2130	584	1346	75
	Ident	22	26	25	26	23	26	22
NC_021312.1 Phaeocystis globosa virus strain 16T	Count	4	9	12	11	3	7	1
	Ident	18	21	26	21	18	18	18
NC_043223.1 Senegalvirus SSV-A contig6 genomic s..	Count	10	18	22	35	9	20	1
	Ident	25	24	25	26	24	24	24
NC_002327.1 Rice grassy stunt virus RNA 5	Count	2	1	6	1	2	7	0
	Ident	23	19	26	19	23	22	0
NC_037667.1 Pandoravirus quercus	Count	65	108	217	221	64	140	10
	Ident	15	26	15	23	13	14	14
NC_001987.1 Ateline herpesvirus 3 complete genome	Count	0	1	2	1	0	1	0
	Ident	0	26	26	26	0	26	0
NC_003624.1 Impatiens necrotic spot virus segmen..	Count	0	1	1	1	1	1	0
	Ident	0	26	26	26	26	26	0
NC_004560.1 Oyster mushroom spherical virus	Count	2	7	11	10	3	6	0
	Ident	17	17	22	26	17	17	0
NC_002051.1 Tomato spotted wilt virus RNA S	Count	1	1	1	1	1	1	1
	Ident	26	26	26	26	26	26	26
NC_019516.1 Cyanophage S-TIM5	Count	0	1	1	1	0	0	0
	Ident	0	26	26	26	0	0	0
NC_002593.1 Plutella xylostella granulovirus	Count	7	14	23	26	8	9	1
	Ident	26	24	21	22	24	22	25
NC_043572.1 Sororoca virus strain BeAr32149 nucl..	Count	54	89	160	176	53	115	4
	Ident	26	22	22	19	26	18	18
NC_002328.1 Rice grassy stunt virus RNA 6	Count	1	0	1	1	1	1	0
	Ident	26	0	26	26	26	26	0
NC_025373.1 Avian paramyxovirus 3 strain turkey/..	Count	0	2	4	2	0	3	0
	Ident	0	20	25	20	0	23	0
NC_063383.1 Monkeypox virus	Count	0	1	1	1	0	0	0
	Ident	0	25	15	25	0	0	0
NC_006658.1 Cotesia congregata virus complete ge..	Count	1	2	3	5	1	3	0
	Ident	25	17	21	17	25	21	0
NC_021148.1 Dill cryptic virus 2 isolate IPP_hor..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	25	0	0	0
NC_009898.1 Paramecium bursaria Chlorella virus ..	Count	1	0	3	6	0	0	0
	Ident	24	0	22	23	0	0	0
NC_006645.1 Cotesia congregata virus complete ge..	Count	4	13	23	24	3	16	1
	Ident	24	24	24	24	24	24	24

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_055426.1 Tacheng Tick Virus 2 strain TC252 nu..	Count	0	0	1	0	0	0	0
	Ident	0	0	24	0	0	0	0
NC_019544.1 Deep-sea thermophilic phage D6E	Count	0	0	0	1	0	0	0
	Ident	0	0	0	24	0	0	0
NC_015874.1 Nam Dinh virus	Count	5	5	13	11	4	9	0
	Ident	24	24	23	24	24	20	0
NC_021097.1 Red clover cryptic virus 2 isolate I..	Count	7	19	32	34	8	20	0
	Ident	24	24	18	21	24	18	0
NC_055411.1 Tomato yellow ring virus isolate TYR..	Count	0	0	1	1	0	0	0
	Ident	0	0	24	13	0	0	0
NC_040606.1 Malacosoma neustria nucleopolyhedrov..	Count	0	2	1	2	0	2	0
	Ident	0	24	12	24	0	12	0
NC_019401.1 Cronobacter phage vB_CsaM_GAP32	Count	2	2	4	5	2	2	0
	Ident	19	16	20	18	20	23	0
NC_055324.1 Cacao virus isolate VP-437R segment M	Count	54	101	174	182	58	127	11
	Ident	21	23	21	23	21	21	21
NC_036599.1 Rice hoja blanca tenuivirus segment ..	Count	10	17	32	26	8	19	3
	Ident	20	20	23	23	20	23	20
NC_038500.1 UNVERIFIED: Rhinolophus associated g..	Count	3	2	5	2	2	6	1
	Ident	23	23	23	23	23	23	23
NC_043612.1 Enseada virus strain 76V-25880 segme..	Count	1	0	1	1	1	2	0
	Ident	23	0	23	23	23	21	0
NC_040681.1 Bufonid herpesvirus 1 strain FO1_2015	Count	1	2	2	1	1	1	0
	Ident	23	18	18	14	23	23	0
NC_047733.1 Synechococcus phage S-RIM8 isolate R..	Count	0	0	6	1	0	2	0
	Ident	0	0	22	21	0	23	0
NC_009240.1 Gryllus bimaculatus nudivirus	Count	7	10	17	17	5	11	2
	Ident	20	23	18	19	17	17	16
NC_028045.1 Tadarida brasiliensis circovirus 1	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_055916.1 Staphylococcus phage LSA2366	Count	3	2	5	3	1	3	1
	Ident	23	23	23	23	23	23	23
NC_047815.1 Erwinia phage vB_EamM_Yoloswag	Count	1	2	3	4	0	0	0
	Ident	14	23	23	23	0	0	0
NC_021099.1 Hop trefoil cryptic virus 2 isolate ..	Count	9	19	43	37	6	26	0
	Ident	15	18	23	19	16	22	0
NC_002816.1 Cydia pomonella granulovirus	Count	0	0	0	1	0	0	0
	Ident	0	0	0	23	0	0	0
NC_008310.2 Hibiscus latent Singapore virus	Count	0	1	4	3	0	2	0
	Ident	0	17	19	22	0	18	0
NC_006553.1 Avian sapelovirus	Count	25	42	76	75	26	58	5
	Ident	22	20	21	19	22	22	21
NC_002520.1 Amsacta moorei entomopoxvirus 'L'	Count	1	1	1	0	1	0	0
	Ident	22	22	22	0	22	0	0
NC_070654.1 Streptococcus phage 7A5	Count	0	1	2	2	0	1	0
	Ident	0	21	22	21	0	21	0
NC_029063.1 Nectarine virus M isolate NeVM/12C51	Count	24	47	79	75	28	50	4
	Ident	21	22	16	16	19	16	17
NC_028250.1 Rosellinia necatrix partitivirus 6 C..	Count	3	8	13	15	3	11	1
	Ident	21	18	20	22	21	20	21
NC_006639.1 Cotesia congregata virus complete ge..	Count	8	15	33	29	7	20	2
	Ident	20	17	19	20	22	16	15
NC_028091.1 Ostreococcus lucimarinus virus 2 iso..	Count	8	12	21	18	7	13	1
	Ident	19	19	19	22	19	19	19
NC_008168.1 Choristoneura fumiferana granulovirus	Count	269	557	965	1006	266	641	30
	Ident	22	22	22	22	21	22	22
NC_009127.1 Cyprinid herpesvirus 3	Count	51	96	171	199	41	115	4
	Ident	20	18	20	20	20	20	22

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_055341.1 Echarte virus segment M	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_007646.1 Ovine herpesvirus 2 strain BJ1035	Count	0	1	0	1	0	1	0
	Ident	0	17	0	17	0	22	0
NC_070664.1 Streptococcus phage CHPC1062	Count	31	71	134	127	26	96	4
	Ident	22	19	20	18	21	18	21
NC_009424.5 Woolly monkey sarcoma virus	Count	0	0	2	0	0	1	0
	Ident	0	0	22	0	0	22	0
NC_004777.1 Yersinia pestis phage phiA1122	Count	4	5	11	17	4	5	1
	Ident	20	20	20	20	20	22	20
NC_043352.1 Glyptapanteles indiensis bracovirus ..	Count	0	1	0	1	0	0	0
	Ident	0	22	0	22	0	0	0
NC_038752.1 RNA4: NCP=non-capsid protein	Count	2	9	12	12	4	7	0
	Ident	21	14	21	14	21	20	0
NC_036600.1 Rosellinia necatrix partitivirus 8 g..	Count	78	143	298	263	67	166	8
	Ident	16	18	20	21	15	17	13
NC_018504.1 Apocheima cinerarium nucleopolyhedro..	Count	5	10	14	13	5	11	0
	Ident	21	20	21	21	21	21	0
NC_040361.1 Planarian secretory cell nidovirus i..	Count	0	4	5	5	0	1	0
	Ident	0	21	20	21	0	21	0
NC_055467.1 Cyclophragma undans nucleopolyhedrov..	Count	2	2	13	6	2	6	0
	Ident	14	14	21	14	14	14	0
NC_023162.1 Carp picornavirus 1 isolate F37/06	Count	1	1	3	1	1	3	0
	Ident	18	18	18	21	18	18	0
NC_001493.2 Ictalurid herpesvirus 1 strain Aubur..	Count	44	102	162	185	45	124	8
	Ident	19	14	21	14	14	17	18
NC_055412.1 Tomato yellow ring virus strain TYRV..	Count	1	2	5	2	0	1	0
	Ident	18	12	21	12	0	14	0
NC_008913.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	2	2	0	0	0
	Ident	0	0	21	21	0	0	0
NC_043326.1 Diolcogaster facetosa bracovirus seg..	Count	7	15	23	23	7	10	2
	Ident	13	20	14	14	21	14	17
NC_036582.1 Flamingopox virus FGPVKD09	Count	0	1	0	1	0	0	0
	Ident	0	21	0	21	0	0	0
NC_054662.1 Streptomyces phage Omar	Count	4	14	27	24	6	15	0
	Ident	16	16	21	16	16	21	0
NC_026440.1 Pandoravirus inopinatum isolate KlaHel	Count	15	36	51	59	11	30	2
	Ident	18	13	16	21	13	17	12
NC_038882.1 Hepatitis C virus strain H77 pCV-H77..	Count	27	48	90	81	18	55	4
	Ident	20	15	17	21	20	17	20
NC_040589.1 Diatom colony associated ssRNA virus..	Count	2	7	13	15	2	10	1
	Ident	21	21	19	21	21	19	21
NC_015280.1 Prochlorococcus phage P-HM1	Count	174	317	644	621	178	411	30
	Ident	21	20	20	20	20	20	21
NC_021095.1 White clover cryptic virus 2 isolate..	Count	1	6	6	8	3	5	0
	Ident	20	19	19	18	20	19	0
NC_008930.1 Glypta fumiferanae ichnovirus segmen..	Count	0	2	1	2	0	0	0
	Ident	0	11	20	11	0	0	0
NC_034557.1 Imjin virus segment M glycoprotein g..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	20	0	0	0
NC_070900.1 Streptococcus phage CHPC926	Count	1	1	1	0	0	0	0
	Ident	20	20	20	0	0	0	0
NC_020867.1 Synechococcus phage S-RIP1 genomic s..	Count	652	1449	2630	2701	662	1815	92
	Ident	20	20	20	20	20	20	17
NC_044937.1 Paramecium bursaria Chlorella virus ..	Count	2	3	13	7	3	9	0
	Ident	20	16	16	16	16	16	0
NC_015960.1 Yoka poxvirus	Count	1	0	3	3	0	1	0
	Ident	20	0	20	20	0	20	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_018875.1 Epinotia aporema granulovirus	Count	3	7	8	11	3	8	1
	Ident	20	16	17	14	15	13	12
NC_053004.1 Salmonella phage TS13	Count	2959	6016	11062	11154	2977	7480	405
	Ident	20	20	20	20	20	20	20
NC_020845.1 Cyanophage MED4-213	Count	12	31	50	55	12	30	4
	Ident	20	19	20	20	19	20	20
NC_039034.1 Common midwife toad ranavirus isolat..	Count	0	4	4	6	2	4	0
	Ident	0	19	18	19	19	20	0
NC_024112.1 Jingmen Tick Virus isolate SY84 segm..	Count	0	3	3	3	0	1	0
	Ident	0	18	17	18	0	20	0
NC_021901.1 Invertebrate iridovirus 22 complete ..	Count	1	0	2	1	1	1	0
	Ident	20	0	17	20	20	20	0
NC_020855.1 Cyanophage P-RSM6 genomic sequence	Count	29	93	128	154	33	105	6
	Ident	19	19	19	19	19	20	16
NC_043318.1 Diolcogaster facetosa bracovirus seg..	Count	13	14	40	40	9	24	2
	Ident	20	20	20	19	20	20	19
NC_008291.1 Taterapox virus	Count	1	3	5	6	2	4	0
	Ident	20	20	20	20	20	20	0
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	10	12	27	24	6	10	0
	Ident	20	20	17	16	15	16	0
NC_011345.1 Agrotis ipsilon multiple nucleopolyh..	Count	3	3	8	3	3	6	0
	Ident	11	20	15	11	20	18	0
NC_004065.1 Murid herpesvirus 1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	20	0
NC_022098.1 Pandoravirus salinus	Count	88	195	306	321	87	204	14
	Ident	14	16	15	16	16	15	20
NC_005902.1 Lymphocystis disease virus - isolate..	Count	4	5	7	6	3	5	1
	Ident	16	19	16	16	16	16	16
NC_009893.1 Tomato yellow dwarf disease associat..	Count	4	6	8	11	5	7	2
	Ident	19	19	19	19	19	18	19
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	7	10	19	17	6	12	1
	Ident	19	19	19	19	19	19	19
NC_043569.1 Iaco virus strain BeAn314206 nucleoc..	Count	0	1	4	1	0	1	0
	Ident	0	19	17	19	0	19	0
NC_009827.1 Hepatitis C virus genotype 6	Count	17	31	59	56	16	37	3
	Ident	17	19	19	19	18	17	17
NC_024697.1 Aureococcus anophagefferens virus is..	Count	13	42	97	86	17	53	3
	Ident	19	14	16	14	19	15	14
NC_003225.3 White spot syndrome virus strain CN01	Count	8	5	14	10	8	12	1
	Ident	19	13	16	16	19	19	13
NC_032274.1 Beihai sesarmid crab virus 4 strain ..	Count	0	0	1	0	0	2	0
	Ident	0	0	19	0	0	19	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	42	116	194	197	41	121	7
	Ident	15	18	18	18	18	19	17
NC_030925.1 Bacillus phage Shbh1	Count	14	21	55	52	16	38	1
	Ident	19	14	16	15	19	15	16
NC_013110.1 Primula malacoides virus China/Mar20..	Count	13	17	46	40	14	28	2
	Ident	19	19	19	19	19	17	19
NC_019491.1 Cyprinid herpesvirus 1 strain NG-J1	Count	8	22	30	36	6	21	0
	Ident	17	15	19	14	17	17	0
NC_043314.1 Diolcogaster facetosa bracovirus clo..	Count	5	6	11	10	3	6	0
	Ident	19	14	15	14	15	19	0
NC_007609.1 Dulcamara mottle virus	Count	6	8	32	24	6	22	0
	Ident	19	16	16	18	15	17	0
NC_004730.1 Indian peanut clump virus RNA 2	Count	0	0	0	1	1	0	0
	Ident	0	0	0	19	19	0	0
NC_021559.1 Prochlorococcus phage P-SSM3 genomic..	Count	0	1	2	1	0	0	0
	Ident	0	19	18	19	0	0	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_012703.1 Nyamanini virus	Count	2	2	4	2	2	6	1
	Ident	19	19	16	14	19	14	14
NC_038838.1 Crimson clover cryptic virus 2 isola..	Count	0	1	2	3	0	0	0
	Ident	0	16	16	19	0	0	0
NC_020100.1 Aspergillus foetidus dsRNA mycovirus..	Count	4	16	21	25	6	18	1
	Ident	16	18	19	18	19	19	19
NC_032104.1 Arachis pintoi virus isolate Var A R..	Count	1	0	1	2	1	1	0
	Ident	19	0	19	15	19	19	0
NC_028099.1 Felis catus gammaherpesvirus 1 isola..	Count	1	1	3	3	0	0	0
	Ident	16	16	19	19	0	0	0
NC_040536.1 Esparto virus isolate SRR3939042_Esp..	Count	4	9	26	20	3	9	1
	Ident	19	16	14	16	16	11	11
NC_028962.1 Staphylococcus phage phiIPLA-C1C	Count	1	2	2	4	0	0	0
	Ident	16	19	12	15	0	0	0
NC_029032.1 Phormidium phage MIS-PhV1A	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_026619.1 Mulberry vein banding virus isolate ..	Count	0	0	1	0	0	1	0
	Ident	0	0	19	0	0	19	0
NC_010989.1 Alternaria alternata dsRNA mycovirus..	Count	7	9	28	26	7	12	1
	Ident	18	19	17	15	18	18	19
NC_019516.2 Cyanophage S-TIM5	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	18	0
NC_071044.1 Bacillus phage vB_BanS_Nate	Count	2	5	3	6	0	2	0
	Ident	18	16	15	18	0	15	0
NC_029573.1 Raspberry leaf blotch virus isolate ..	Count	1	1	1	1	1	1	1
	Ident	18	18	18	18	18	18	18
NC_043307.1 Diolcogaster facetosa bracovirus seg..	Count	28	39	76	69	28	63	5
	Ident	15	15	13	18	15	15	15
NC_031338.1 Moku virus isolate Big Island	Count	5	17	24	27	6	11	1
	Ident	18	18	18	18	18	18	18
NC_006151.1 Suid herpesvirus 1	Count	1	1	2	2	2	3	0
	Ident	17	17	17	17	17	18	0
NC_008200.1 Mycobacterium phage PLot	Count	1	8	7	11	2	5	0
	Ident	14	18	18	18	14	14	0
NC_013015.1 Sclerotinia sclerotiorum partitiviru..	Count	0	1	1	1	0	1	0
	Ident	0	17	18	17	0	16	0
NC_043351.1 Glyptapanteles indiensis bracovirus ..	Count	0	1	1	3	0	2	0
	Ident	0	18	18	12	0	18	0
NC_013220.1 Phytophthora infestans RNA virus 1 R..	Count	4	7	16	18	4	11	1
	Ident	15	18	16	15	15	15	14
NC_015492.1 Grapevine Bulgarian latent virus seg..	Count	0	0	1	1	0	0	0
	Ident	0	0	18	17	0	0	0
NC_008912.1 Glypta fumiferanae ichnovirus segmen..	Count	1	1	1	1	1	1	1
	Ident	17	17	17	17	17	17	17
NC_043403.1 Una virus non structural polyprotein..	Count	0	1	4	2	0	2	0
	Ident	0	16	17	17	0	16	0
NC_001632.1 Rice tungro spherical virus	Count	0	1	2	1	0	1	0
	Ident	0	12	15	12	0	17	0
NC_070670.1 Streptococcus phage CHPC595	Count	2	2	5	3	3	5	1
	Ident	16	16	16	16	16	17	16
NC_048072.1 Streptomyces phage Darolandstone	Count	1	7	9	10	2	8	0
	Ident	17	17	17	17	17	17	0
NC_039203.1 Polygonum ringspot tospovirus isolat..	Count	1	0	1	0	1	2	0
	Ident	14	0	14	0	14	17	0
NC_055231.1 Orthopoxvirus Abatino	Count	0	0	0	1	0	0	0
	Ident	0	0	0	17	0	0	0
NC_020878.1 Prochlorococcus phage P-GSP1 genomic..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	17	0	0	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_055135.1 Retroperitoneal fibromatosis-associa..	Count	1	5	6	8	1	4	0
	Ident	13	13	17	13	13	13	0
NC_029035.2 Colombian potato soil-borne virus RN..	Count	0	0	1	1	0	0	0
	Ident	0	0	17	17	0	0	0
NC_038825.1 Flammulina velutipes browning virus ..	Count	6	18	23	35	7	15	1
	Ident	17	16	17	16	16	17	17
NC_016447.1 Aotine herpesvirus 1 strain S34E	Count	30	59	101	129	35	79	4
	Ident	15	15	15	17	15	15	15
NC_043508.1 Persea americana chrysovirus segment..	Count	2	4	6	7	2	4	2
	Ident	15	15	17	17	15	15	15
NC_020474.2 Elephantid herpesvirus 1	Count	0	0	1	1	0	0	0
	Ident	0	0	17	17	0	0	0
NC_070989.1 Escherichia phage vB_EcoP-CHD5UKE1	Count	1	2	3	3	1	1	0
	Ident	16	17	16	17	16	16	0
NC_008962.1 Hyposoter fugitivus ichnovirus segme..	Count	8	11	16	19	8	15	0
	Ident	16	17	16	17	16	16	0
NC_009448.2 Saffold virus	Count	0	1	5	3	0	3	0
	Ident	0	17	14	17	0	17	0
NC_024474.1 Pigeon adenovirus 1 complete genome	Count	10	16	25	31	10	17	1
	Ident	15	15	12	17	15	15	15
NC_016448.1 Saimiriine herpesvirus 4 strain SqSHV	Count	1	0	1	0	1	1	0
	Ident	17	0	17	0	17	17	0
NC_047813.1 Staphylococcus phage Andhra	Count	170	315	605	611	189	407	27
	Ident	17	17	17	17	17	17	16
NC_029800.1 Iris yellow spot virus non-structura..	Count	2	8	17	15	5	8	0
	Ident	17	17	17	17	17	17	0
NC_002687.1 Ectocarpus siliculosus virus 1	Count	0	1	1	2	0	0	0
	Ident	0	17	17	15	0	0	0
NC_008862.1 Glypta fumiferanae ichnovirus segmen..	Count	0	1	1	1	0	2	0
	Ident	0	17	17	17	0	13	0
NC_043331.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	1	0	0	1	0
	Ident	0	0	16	0	0	16	0
NC_008908.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	1	1	0	0
	Ident	0	0	16	16	16	0	0
NC_006650.1 Cotesia congregata virus complete ge..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	16	0	0	0
NC_038378.1 Cacao swollen shoot CD virus isolate..	Count	11	22	42	35	9	27	0
	Ident	15	16	14	15	13	14	0
NC_037660.1 Botrytis cinerea fusarivirus 1	Count	1	0	2	2	0	1	0
	Ident	16	0	16	16	0	16	0
NC_006651.1 Cotesia congregata virus complete ge..	Count	1	1	2	0	0	2	0
	Ident	16	16	16	0	0	14	0
NC_024502.1 Gentian ovary ring-spot virus genomi..	Count	24	42	92	86	21	63	1
	Ident	16	14	16	16	16	16	15
NC_033829.1 Kallithea virus isolate DrosEU46_Kha..	Count	8	16	16	22	8	18	2
	Ident	14	14	13	13	14	13	16
NC_037666.1 Pandoravirus neocaledonia	Count	59	122	218	212	54	139	7
	Ident	14	15	15	15	14	16	12
NC_020235.1 Rosellinia necatrix partitivirus 2 C..	Count	18	27	62	55	20	37	3
	Ident	16	14	15	16	16	12	16
NC_033102.1 Hubei odonate virus 3 strain QTM2515..	Count	1	0	1	0	1	0	0
	Ident	16	0	16	0	16	0	0
NC_071130.1 Klebsiella phage BUCT_47333	Count	1	1	4	3	2	2	0
	Ident	12	15	14	15	16	11	0
NC_055235.1 Baboon cytomegalovirus OCOM4-37	Count	1	1	0	1	1	0	0
	Ident	16	16	0	16	16	0	0
NC_021929.1 Malvastrum leaf curl deltasatellite ..	Count	0	0	2	0	0	2	0
	Ident	0	0	15	0	0	16	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_025381.1 Hibiscus latent Fort Pierce virus ge..	Count	0	0	1	1	0	0	0
	Ident	0	0	14	16	0	0	0
NC_008907.1 Glypta fumiferanae ichnovirus segmen..	Count	4	3	6	6	2	4	1
	Ident	13	16	13	16	16	16	16
NC_016072.1 Megavirus chiliensis	Count	0	0	2	1	0	0	0
	Ident	0	0	16	15	0	0	0
NC_001798.2 Human herpesvirus 2 strain HG52	Count	0	1	0	1	1	0	0
	Ident	0	16	0	16	16	0	0
NC_033778.1 Leptopilina boulardi filamentous vir..	Count	1	1	2	1	1	1	1
	Ident	12	12	16	12	12	12	12
NC_009816.1 Corynebacterium phage P1201	Count	0	1	1	1	0	0	0
	Ident	0	16	16	16	0	0	0
NC_029594.1 Lake Sarah-associated circular virus..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	16	0
NC_023015.1 Hedyotis uncinella yellow mosaic bet..	Count	1	0	1	1	0	1	0
	Ident	16	0	16	16	0	16	0
NC_038840.1 Heterobasidion partitivirus 2 isolat..	Count	0	0	1	1	0	1	0
	Ident	0	0	16	14	0	16	0
NC_049942.1 Escherichia phage JLK-2012	Count	1	0	4	4	1	3	0
	Ident	14	0	15	16	14	14	0
NC_007241.1 Vicia cryptic virus RNA1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	16	0	0	0
NC_021858.1 Pandoravirus dulcis	Count	68	126	222	251	63	148	5
	Ident	15	15	15	15	16	16	15
NC_011421.1 Bacillus phage SPO1	Count	25	43	90	99	20	59	3
	Ident	15	14	13	14	12	15	12
NC_007001.1 Cassia yellow blotch virus RNA3	Count	0	0	4	1	0	1	0
	Ident	0	0	13	15	0	13	0
NC_031008.1 Staphylococcus phage SLPW	Count	0	0	1	0	0	0	0
	Ident	0	0	15	0	0	0	0
NC_038290.1 Watermelon bud necrosis virus strain..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	15	0	0	0
NC_055359.1 Odrenisrou virus segment M	Count	0	1	1	1	0	0	0
	Ident	0	15	15	15	0	0	0
NC_040534.1 Yichang virus isolate HB-MLV	Count	0	1	2	1	1	1	0
	Ident	0	15	15	15	15	15	0
NC_028251.1 Rosellinia necatrix partitivirus 6 R..	Count	0	0	1	0	0	1	0
	Ident	0	0	15	0	0	15	0
NC_001348.1 Human herpesvirus 3	Count	5	4	8	10	2	5	1
	Ident	15	15	15	15	15	15	15
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	576	1194	2123	2176	584	1386	86
	Ident	15	15	15	15	15	15	15
NC_033436.1 Wuchan romanomermis nematode virus 2..	Count	0	1	0	1	0	0	0
	Ident	0	15	0	15	0	0	0
NC_043292.1 Cotesia glomerata bracovirus putativ..	Count	1	3	3	3	1	1	0
	Ident	15	15	12	12	15	15	0
NC_037665.1 Pandoravirus macleodensis	Count	233	522	885	922	218	597	35
	Ident	15	15	14	15	14	15	15
NC_049489.1 Arthrobacter phage DrYang	Count	1	1	3	1	1	4	0
	Ident	15	15	15	15	15	15	0
NC_010276.1 Orgyia leucostigma NPV	Count	0	2	0	3	0	1	0
	Ident	0	14	0	15	0	15	0
NC_026283.1 Maprik virus isolate MK7532 segment M	Count	1	1	2	2	1	1	0
	Ident	15	15	15	15	15	15	0
NC_049969.1 Bacillus phage DK3	Count	0	1	2	2	0	1	0
	Ident	0	15	15	15	0	15	0
NC_022617.1 Red clover cryptic virus 1 isolate I..	Count	3	10	12	15	4	8	0
	Ident	12	12	14	12	15	13	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_043313.1 Diolcogaster facetosa bracovirus clo..	Count	3	10	14	18	5	11	1
	Ident	13	14	14	14	15	15	13
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	22	27	60	57	15	38	2
	Ident	15	15	15	15	15	15	14
NC_055364.1 Gordil virus isolate Dak ANBr 496d s..	Count	0	0	0	1	0	1	0
	Ident	0	0	0	14	0	15	0
NC_038553.1 Heterosigma akashiwo virus 01 isolat..	Count	13	40	55	58	14	32	1
	Ident	13	15	14	14	15	13	12
NC_020871.1 Listeria phage vB_LmoM_AG20	Count	6	6	18	8	5	14	0
	Ident	15	15	15	15	15	15	0
NC_020231.1 Caviid herpesvirus 2 strain 21222	Count	10	25	43	48	12	31	4
	Ident	15	12	13	14	15	13	14
NC_001672.1 Tick-borne encephalitis virus	Count	1	1	1	5	0	1	0
	Ident	15	15	15	13	0	15	0
NC_049372.1 Roseobacter phage RD-1410W1-01	Count	2	1	6	3	2	2	0
	Ident	14	15	14	14	14	14	0
NC_020101.1 Aspergillus foetidus dsRNA mycovirus..	Count	0	0	1	0	0	1	0
	Ident	0	0	15	0	0	15	0
NC_006656.1 Cotesia congregata virus complete ge..	Count	4	7	11	18	5	8	0
	Ident	14	14	14	15	14	11	0
NC_043566.1 Anhembi virus strain SPAr2984 nucleo..	Count	1	5	9	7	1	6	0
	Ident	15	15	15	15	15	11	0
NC_038332.1 Fowl adenovirus 6 strain CR119	Count	0	1	3	3	2	0	0
	Ident	0	15	13	14	13	0	0
NC_004122.1 Spring beauty latent virus RNA 3	Count	1	0	3	2	1	2	0
	Ident	14	0	14	14	14	14	0
NC_022332.1 Eel picornavirus 1 strain F15/05	Count	5	9	17	23	6	13	1
	Ident	12	12	13	14	12	12	12
NC_049835.1 Klebsiella phage vB_KpnS_Domnhall	Count	1	4	4	5	0	3	0
	Ident	14	14	14	14	0	14	0
NC_004067.1 Pepino mosaic virus	Count	1	0	1	0	1	1	0
	Ident	14	0	14	0	14	14	0
NC_023681.1 Mamestra brassicae MNPV strain K1	Count	0	1	5	3	1	8	0
	Ident	0	14	14	14	14	14	0
NC_048651.1 Aeribacillus phage AP45	Count	639	1260	2342	2353	627	1559	101
	Ident	14	14	14	14	14	14	14
NC_006659.1 Cotesia congregata virus complete ge..	Count	16	27	48	39	10	34	5
	Ident	14	14	14	13	13	14	13
NC_055142.1 Lymphocryptovirus Macaca/pfe-lcl-E3	Count	0	3	7	6	0	3	0
	Ident	0	13	14	14	0	14	0
NC_010356.1 Glossina pallidipes salivary gland h..	Count	0	3	2	5	2	0	0
	Ident	0	14	13	13	12	0	0
NC_043100.1 Bat astrovirus Tm/Guangxi/LD77/2007 ..	Count	0	0	1	0	0	2	0
	Ident	0	0	13	0	0	14	0
NC_038842.1 Heterobasidion partitivirus 8 strain..	Count	0	1	2	1	0	1	0
	Ident	0	14	14	14	0	14	0
NC_010537.1 Acidianus filamentous virus 9	Count	0	0	2	0	0	3	0
	Ident	0	0	14	0	0	12	0
NC_043325.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	14	0
NC_023760.1 Mink coronavirus strain WD1127	Count	1	0	0	0	0	0	0
	Ident	14	0	0	0	0	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	228	493	852	859	224	510	31
	Ident	14	14	14	14	14	13	14
NC_043328.1 Diolcogaster facetosa bracovirus seg..	Count	1	2	7	6	1	4	0
	Ident	14	12	13	12	14	14	0
NC_003084.1 Culex nigripalpus NPV	Count	0	0	1	0	0	2	0
	Ident	0	0	14	0	0	14	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_053235.1 Gordonia phage JKSyngboy	Count	3	3	12	8	2	5	0
	Ident	14	14	14	14	14	14	0
NC_038833.1 Heterobasidion partitivirus 15 strai..	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_006960.1 Pleurotus ostreatus virus 1 RNA-2	Count	0	2	0	3	0	2	0
	Ident	0	13	0	14	0	14	0
NC_008929.1 Glypta fumiferanae ichnovirus segmen..	Count	11	28	54	47	13	41	3
	Ident	14	14	14	14	14	14	14
NC_038932.1 Lychnis ringspot virus RNA for beta-A	Count	1	3	3	4	1	2	0
	Ident	11	14	12	13	11	13	0
NC_024298.1 Sclerotinia sclerotiorum debilitatio..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	14	0
NC_007242.1 Vicia cryptic virus RNA2	Count	2	3	6	5	2	4	0
	Ident	14	14	14	14	14	14	0
NC_038356.1 Torque teno midi virus 9 DNA	Count	0	0	0	1	0	0	0
	Ident	0	0	0	14	0	0	0
NC_002512.2 Rat cytomegalovirus Maastricht	Count	39	86	142	155	34	83	10
	Ident	11	13	13	11	13	13	13
NC_010991.1 Alternaria alternata dsRNA mycovirus..	Count	0	1	0	1	0	0	0
	Ident	0	13	0	13	0	0	0
NC_007151.1 Chrysodeixis chalcites nucleopolyhed..	Count	2	2	7	8	3	5	0
	Ident	13	13	13	13	13	13	0
NC_043574.1 Cachoeira Porteira virus strain BeAr..	Count	0	1	1	1	0	0	0
	Ident	0	13	13	13	0	0	0
NC_000852.5 Paramecium bursaria Chlorella virus 1	Count	0	1	1	1	0	1	0
	Ident	0	13	13	13	0	13	0
NC_019501.1 Enterobacteria phage IME10	Count	0	1	0	1	0	0	0
	Ident	0	13	0	13	0	0	0
NC_002645.1 Human coronavirus 229E	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	13	0
NC_037663.1 Botrytis cinerea hypovirus 1 satelli..	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_038933.1 Lychnis ringspot virus RNA for gamma..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	13	0	0	0
NC_037664.1 Botrytis cinerea hypovirus 1 satelli..	Count	0	1	0	1	0	0	0
	Ident	0	13	0	13	0	0	0
NC_006657.1 Cotesia congregata virus complete ge..	Count	1	0	2	2	1	5	0
	Ident	13	0	13	13	13	13	0
NC_023021.1 Formica exsecta virus 1 isolate Fex1	Count	4	1	5	4	4	3	0
	Ident	12	11	13	12	12	12	0
NC_020234.1 Rosellinia necatrix partitivirus 2 R..	Count	2	2	8	5	2	9	0
	Ident	13	13	13	13	13	13	0
NC_006633.1 Cotesia congregata virus complete ge..	Count	0	0	2	0	0	1	0
	Ident	0	0	13	0	0	11	0
NC_007021.1 Staphylococcus phage Twort	Count	0	1	1	1	0	1	0
	Ident	0	13	13	13	0	13	0
NC_011038.1 Yersinia phage Yepe2	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	13	0
NC_055862.1 Vibrio phage ValB1MD-2	Count	1	6	13	11	0	7	0
	Ident	13	13	13	13	0	13	0
NC_010646.1 Beluga Whale coronavirus SW1	Count	0	0	1	0	0	1	0
	Ident	0	0	13	0	0	13	0
NC_023684.1 Rhizoctonia solani dsRNA virus 2 seg..	Count	0	0	1	0	0	1	0
	Ident	0	0	13	0	0	13	0
NC_048751.1 Pantoea phage vB_PagM_LIET2	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_003835.1 Tulare apple mosaic virus RNA3	Count	1	1	1	1	1	1	1
	Ident	13	13	13	13	13	13	13

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_038832.1 Heterobasidion partitivirus 13 strai..	Count	0	0	2	1	0	2	0
	Ident	0	0	13	12	0	12	0
NC_024209.1 Mycobacterium phage Hawkeye	Count	1	0	1	1	1	1	0
	Ident	13	0	13	13	13	13	0
NC_055297.1 Alstroemeria necrotic streak virus i..	Count	3	1	10	5	1	3	0
	Ident	13	13	13	13	13	13	0
NC_049392.1 Escherichia phage ESSI2_ev239 genome..	Count	9	19	28	27	8	18	1
	Ident	12	12	12	13	12	12	12
NC_038352.1 Torque teno midi virus 5 DNA	Count	0	0	1	0	0	0	0
	Ident	0	0	13	0	0	0	0
NC_030200.1 Macaca nemestrina herpesvirus 7	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0
NC_007993.1 Campoletis sonorensis ichnovirus seg..	Count	0	2	2	3	1	0	0
	Ident	0	12	12	12	12	0	0
NC_048875.1 Pantoea phage vB_PagM_SSEM1	Count	0	2	2	2	0	0	0
	Ident	0	12	12	12	0	0	0
NC_038977.1 Sclerotinia sclerotiorum deltaflexiv..	Count	0	0	0	1	0	1	0
	Ident	0	0	0	12	0	12	0
NC_023627.1 Laodelphax striatella honeydew virus..	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0
NC_048798.1 Klebsiella phage Marfa	Count	6	10	23	18	8	22	0
	Ident	12	12	12	12	12	12	0
NC_028491.1 Diatraea saccharalis granulovirus	Count	0	1	3	3	0	0	0
	Ident	0	12	12	12	0	0	0
NC_006662.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_055577.1 Physalis rugose mosaic virus isolate..	Count	1	0	3	2	1	2	0
	Ident	12	0	12	12	12	11	0
NC_052978.1 Proteus phage Saba	Count	35	52	110	107	37	81	7
	Ident	12	12	12	12	12	12	12
NC_043346.1 Glyptapanteles indiensis bracovirus ..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_030953.1 Shigella phage SHFML-11	Count	2	1	5	1	1	2	0
	Ident	12	11	12	11	12	12	0
NC_022614.1 Dill cryptic virus 1 isolate IPP_hor..	Count	1	0	2	1	1	1	0
	Ident	12	0	12	12	12	12	0
NC_055020.1 Staphylococcus phage SN8	Count	1	0	1	1	1	1	0
	Ident	12	0	12	12	12	12	0
NC_018874.1 Abalone herpesvirus Victoria/AUS/2009	Count	3	7	12	11	3	5	1
	Ident	12	12	12	12	12	12	12
NC_022896.1 Sclerotinia sclerotiorum hypovirus 2..	Count	0	1	0	1	0	0	0
	Ident	0	12	0	12	0	0	0
NC_025474.1 Crohivirus A gene for polyprotein	Count	1	1	2	1	0	2	0
	Ident	12	12	12	11	0	12	0
NC_010990.1 Alternaria alternata dsRNA mycovirus..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	12	0
NC_040456.1 Medicago sativa alphapartitivirus 1 ..	Count	2	3	7	9	1	5	0
	Ident	12	12	12	12	12	12	0
NC_044938.1 Heliothis virescens ascovirus 3f iso..	Count	0	0	1	1	0	0	0
	Ident	0	0	12	12	0	0	0
NC_023442.1 Buzura suppressaria nucleopolyhedrov..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	12	0	0	0
NC_024150.1 California sea lion adenovirus 1 str..	Count	0	1	0	1	0	0	0
	Ident	0	12	0	12	0	0	0
NC_003542.1 Cowpea chlorotic mottle virus RNA 3	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	12	0
NC_011349.1 Seneca valley virus	Count	1	0	1	1	1	1	0
	Ident	12	0	12	12	12	12	0

Table 5.4 Viral genomes mapping to the CALIBER Nettle dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_020102.1 Aspergillus foetidus dsRNA mycovirus..	Count	0	0	2	0	0	2	0
	Ident	0	0	12	0	0	12	0
NC_038295.1 Fathead minnow nidovirus replicase p..	Count	13	15	33	32	11	27	1
	Ident	11	11	11	11	11	11	11
NC_024114.1 Jingmen Tick Virus isolate SY84 segm..	Count	1	1	1	1	1	1	1
	Ident	11	11	11	11	11	11	11
NC_038426.1 Hepacivirus B polypeptide gene	Count	0	1	1	1	0	0	0
	Ident	0	11	11	11	0	0	0
NC_043317.1 Diolcogaster facetosa bracovirus seg..	Count	0	1	2	1	0	2	0
	Ident	0	11	11	11	0	11	0
NC_038318.1 Mouse Mosavirus strain Mosa.M-7 poly..	Count	0	0	1	0	0	1	0
	Ident	0	0	11	0	0	11	0
NC_008725.1 Maruca vitrata MNPV	Count	0	2	1	2	0	0	0
	Ident	0	11	11	11	0	0	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	5	8	16	19	5	9	1
	Ident	11	11	11	11	11	11	11
NC_038957.1 Picornavirus HK21 polyprotein gene	Count	0	1	1	1	0	0	0
	Ident	0	11	11	11	0	0	0
NC_055497.1 Passiflora edulis symptomless virus ..	Count	0	0	1	0	0	1	0
	Ident	0	0	11	0	0	11	0
NC_004015.1 Sorghum chlorotic spot virus RNA 2	Count	0	0	1	0	0	1	0
	Ident	0	0	11	0	0	11	0
NC_025480.2 Carrot torradovirus 1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	11	0	0	0
NC_014602.1 Raspberry latent virus segment S5	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	11	0
NC_030131.1 Duck faeces associated circular DNA ..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	11	0	0	0
NC_038839.1 Heterobasidion partitivirus 2 isolat..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	11	0
NC_021923.1 Hemileuca sp. nucleopolyhedrovirus	Count	1	0	3	3	1	2	0
	Ident	11	0	11	11	11	11	0
NC_015253.1 Brochothrix phage A9	Count	0	3	3	4	0	3	0
	Ident	0	11	11	11	0	11	0
NC_023675.1 Porcine astrovirus 4 strain 35/USA	Count	4	3	9	11	2	6	0
	Ident	11	11	11	11	11	11	0
Novel mapping*	Count	9071	18654	33828	34547	8955	22404	1289
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

Fowkes Pea datasets

These datasets have previously been analysed by Fowkes et al. (Fowkes et al., 2021) utilising an assembly- and alignment-based tool, Angua3. Their analysis for these bulked-sample datasets included computational detection of plant-infecting viral genomes, as well as confirmation of candidate viruses by real-time RT-PCR. These previous results were able to act as a positive control for our genome-mapping analysis, an effective virus-detection tool must at least be able to find confirmed viral genomes, but may find additional ones. Results are summarised in Tables 5.5-5.7.

Fowkes Pea 14 was confirmed to contain Turnip yellows virus at a high (68.08-99.83%) estimated incidence, Pea enation mosaic virus-2 at a low (3.49-14.07%) incidence, and Soybean dwarf virus at a very low (0.21-5.16%) incidence. All of these genomes were mapped to by reads detected by all tools in Table 5.5. Turnip yellows virus had many hits mapped with a remarkably similar number of reads across tools (509-595) with the majority being unanimous in assignment (436), and a high identity (93%). There were additional *Polerovirus* mappings, including Brassica yellows virus isolate BrYV-ABJ, which had similar numbers of hits (361-442) as Turnip yellows virus but with a slightly higher median identity (93-94%). Brassica yellows virus is known to be very closely related to Turnip yellows virus, with a high rate of recombination and observation of paraphyly between the two, which has led some to consider Brassica yellows virus to be a variant of Turnip yellows virus (Peng et al., 2023) (Sõmera et al., 2021). Other *Polerovirus* detected in this dataset include Faba bean polerovirus 1 strain 5253, Beet western yellows virus, Beet mild yellowing virus, Beet chlorosis virus, and further mappings at low counts or identities. Consistent hit counts and a high proportion of unanimity were observed for these genomes. These hits confirm the presence of the positive control of Turnip yellows virus in this bulk sample. The high identity mapping of multiple genomes may represent a single population of highly recombinant *Polerovirus* genomes across the site that the samples were taken from. This would be consistent with the high rate of genomic crossover in this genus, which is known to generate novel viral strains and species (LaTourrette et al., 2021). Pea enation mosaic virus-2 had lower hit counts (54-66), but also showed a high proportion unanimous (46) and high identity (92%). Soybean dwarf virus had a very low hit count (4-5), consistent with the very low incidence reported by Fowkes et al. The unanimous mapping fraction was as large as the smallest tool hit count, representing an overlap coefficient of 1 between all tools.

The only additional hit was by DeepVirFinder. This high unanimity and a relatively high identity (86-89%), despite low read numbers, points to a low titre presence in the bulk dataset. Outside of positive control genomes, some additional genomes showed high mapping counts. Non-plant-infecting viral genomes with highly variable hit counts and low unanimity made up most of these. Interestingly, Escherichia phage phiX174 showed a high median mapping identity (93-95%), even for the numerous (3180) DeepVirFinder hits, which was uncommon for this set of genomes. Determining what these hits may represent would require further analysis into the exact regions they fall into, the lengths and features of contigs produced from their assembly,

and any open reading frames present in them. Additionally, this dataset presented an unmapped fraction with highly varying hits counts (38-970).

Fowkes Pea 20 had a confirmed presence of Turnip yellows virus (16.27-42.99% estimated incidence), Pea enation mosaic virus-1 (21.63-56.58% incidence), Pea enation mosaic virus-2 (23.55-62.15% incidence), and Pea enation mosaic virus satellite RNA (no incidence estimate); Fowkes Pea 15 showed the presence of Turnip yellows virus (1.01-8.43% incidence), Pea enation mosaic virus-1 (8.93-26.56% incidence), Pea enation mosaic virus-2 (0.63-7.67% incidence), and Soybean dwarf virus (1.51-9.41% incidence). Our read mapping analysis for these datasets is summarised in Tables 5.6 and 5.7, respectively. All expected genomes had mapped hits in both datasets, with hit counts broadly following incidence. Some genomes had a very high hit count compared to relative incidence, such as Pea enation mosaic virus-1 in the Fowkes Pea 15 dataset (3970-4063 reads, Table 5.7), and others a very low hit count, such as Turnip yellows virus in the same dataset (2 reads). As incidence is a measure of how many individual samples in the bulked sample are expected to show any presence of a viral genome, this would indicate a difference in viral titre for these genomes. The Fowkes Pea 20 dataset (Table 5.6), similar to previous Fowkes Pea 14 dataset (Table 5.5), included a bacteriophage genome, Proteus phage VB_PmiS-Isfahan, with a very high median mapping identity (99% identity), wide range of hit counts (240-11647), and very low unanimity (3 reads). The Fowkes Pea 15 dataset (Table 5.7), on the other hand, showed this pattern for soybean dwarf virus (97% identity; 381-16878 hits; 3 unanimous), which was a known positive control for this dataset. Both datasets also showed large unmapped fractions with low unanimity.

Table 5.5 Viral genomes mapping to the Fowkes Pea 14 dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_001422.1 Escherichia phage phiX174	Count	137	889	3180	1039	209	2096	27
	Ident	94	94	93	94	95	93	93
NC_055495.1 Faba bean polerovirus 1 strain 5253	Count	91	92	75	91	77	87	60
	Ident	94	94	94	94	94	94	94
NC_004756.1 Beet western yellows virus	Count	33	33	31	33	30	32	28
	Ident	94	94	94	94	94	94	94
NC_016038.2 Brassica yellows virus isolate BrYV-..	Count	442	441	361	440	373	433	314
	Ident	94	94	93	94	94	94	93
NC_003743.1 Turnip yellows virus	Count	593	595	514	591	509	581	436
	Ident	93	93	93	93	93	93	93
NC_003491.1 Beet mild yellowing virus	Count	66	66	56	64	54	65	46
	Ident	93	93	93	93	93	93	92
NC_003853.1 Pea enation mosaic virus-2	Count	69	68	54	70	58	69	46
	Ident	93	93	93	93	93	92	92
NC_003056.1 Soybean dwarf virus genomic RNA	Count	4	4	5	4	4	4	4
	Ident	89	89	86	89	89	89	89
NC_002766.1 Beet chlorosis virus	Count	23	23	22	23	20	25	17
	Ident	88	88	85	88	87	87	87
NC_034246.1 Cowpea polerovirus 1 isolate BE167	Count	1	1	0	1	1	1	0
	Ident	84	84	0	84	84	84	0
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	29	170	606	200	36	441	8
	Ident	76	76	76	76	76	76	76
NC_024382.1 Alcelaphine herpesvirus 2 isolate to..	Count	0	0	1	0	0	0	0
	Ident	0	0	68	0	0	0	0
NC_029302.1 Piscine myocarditis-like virus isola..	Count	0	1	5	1	0	0	0
	Ident	0	66	66	66	0	0	0
NC_048049.1 Synechococcus phage S-T4	Count	21	124	493	160	21	280	7
	Ident	55	55	64	55	55	55	55
NC_016072.1 Megavirus chiliensis	Count	0	0	1	0	0	0	0
	Ident	0	0	58	0	0	0	0
NC_010809.1 Melon aphid-borne yellows virus	Count	1	1	1	1	1	1	1
	Ident	57	57	57	57	57	57	57
NC_021484.1 Maize yellow dwarf virus-RMV	Count	2	2	2	2	1	2	1
	Ident	35	35	35	35	54	35	54
NC_034207.1 African eggplant yellowing virus iso..	Count	3	3	2	3	3	3	2
	Ident	50	50	36	50	50	50	36
NC_029691.1 Ixeridium yellow mottle virus 1 isol..	Count	2	2	2	1	1	2	1
	Ident	48	48	48	48	48	48	48
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	4	44	120	65	10	88	0
	Ident	33	13	13	12	48	13	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	1	2	6	2	1	3	1
	Ident	45	38	38	38	45	39	45
NC_031032.1 Bacillus phage Stitch	Count	0	0	1	0	0	0	0
	Ident	0	0	45	0	0	0	0
NC_001782.1 Saccharomyces cerevisiae killer viru..	Count	0	0	1	0	0	0	0
	Ident	0	0	42	0	0	0	0
NC_036582.1 Flamingopox virus FGPVKD09	Count	0	0	2	0	0	1	0
	Ident	0	0	40	0	0	34	0
NC_001747.1 Potato leafroll virus	Count	1	9	35	14	1	24	1
	Ident	39	35	39	35	39	34	39
NC_048171.1 Synechococcus phage S-B28	Count	0	0	2	0	0	0	0
	Ident	0	0	38	0	0	0	0
NC_048004.1 Salmonella phage 3A_8767	Count	0	1	1	1	0	0	0
	Ident	0	37	37	37	0	0	0
NC_038828.1 Heterobasidion RNA virus 1 isolate H..	Count	0	0	2	0	0	0	0
	Ident	0	0	34	0	0	0	0

Table 5.5 Viral genomes mapping to the Fowkes Pea 14 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_004102.1 Hepatitis C virus genotype 1	Count	1	3	8	2	1	6	0
	Ident	21	30	34	30	21	33	0
NC_014545.1 Cotton leafroll dwarf virus	Count	1	1	1	1	1	1	1
	Ident	34	34	34	34	34	34	34
NC_034265.1 Tobacco virus 2	Count	1	6	17	9	2	14	1
	Ident	24	33	24	33	33	33	24
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	2	8	41	9	2	33	0
	Ident	21	33	33	33	21	26	0
NC_030230.1 Tokyovirus A1 DNA	Count	0	5	13	6	0	9	0
	Ident	0	26	33	26	0	18	0
NC_020864.1 Micromonas pusilla virus 12T genomic..	Count	0	0	1	0	0	0	0
	Ident	0	0	33	0	0	0	0
NC_030225.1 Pepo aphid-borne yellows virus isola..	Count	1	1	0	1	1	1	0
	Ident	33	33	0	33	33	33	0
NC_009823.1 Hepatitis C virus genotype 2	Count	2	15	73	19	3	45	1
	Ident	18	18	32	18	18	26	18
NC_008586.1 Ecotropis obliqua NPV	Count	0	0	1	0	0	1	0
	Ident	0	0	31	0	0	31	0
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_008168.1 Choristoneura fumiferana granulovirus	Count	13	62	215	75	11	121	4
	Ident	30	24	26	22	30	17	30
NC_043054.1 Bubaline alphaherpesvirus 1 strain b6	Count	99	722	2457	854	145	1585	20
	Ident	29	29	23	29	29	22	29
NC_040743.1 Alstroemeria yellow spot virus isola..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	26	0	0	0
NC_025412.1 Melbournevirus isolate 1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	26	0
NC_032255.1 Plodia interpunctella granulovirus i..	Count	0	0	2	0	0	1	0
	Ident	0	0	26	0	0	26	0
NC_014637.1 Cafeteria roenbergensis virus BV-PW1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	25	0
NC_021099.1 Hop trefoil cryptic virus 2 isolate ..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	25	0
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	0	0	4	0	0	1	0
	Ident	0	0	24	0	0	19	0
NC_018072.1 Bean necrotic mosaic virus segment M	Count	0	0	1	0	0	0	0
	Ident	0	0	24	0	0	0	0
NC_053004.1 Salmonella phage TS13	Count	46	329	1148	413	68	729	9
	Ident	22	21	21	21	21	21	21
NC_020867.1 Synechococcus phage S-RIP1 genomic s..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	21	0
NC_048651.1 Aeribacillus phage AP45	Count	1	8	28	10	2	28	0
	Ident	21	21	21	14	21	13	0
NC_070848.1 Vibrio phage BUCT194	Count	0	1	1	1	0	1	0
	Ident	0	20	20	20	0	20	0
NC_008310.2 Hibiscus latent Singapore virus	Count	1	1	1	1	1	1	1
	Ident	20	20	20	20	20	20	20
NC_020855.1 Cyanophage P-RSM6 genomic sequence	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_044937.1 Paramecium bursaria Chlorella virus ..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	0	0	2	0	0	2	0
	Ident	0	0	19	0	0	14	0
NC_043223.1 Senegalvirus SSV-A contig6 genomic s..	Count	0	1	3	1	0	0	0
	Ident	0	19	19	19	0	0	0
NC_024486.1 Duck adenovirus 2 strain GR	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0

Table 5.5 Viral genomes mapping to the Fowkes Pea 14 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	1	11	48	17	4	38	0
	Ident	17	14	17	16	17	17	0
NC_043313.1 Diolcogaster facetosa bracovirus clo..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_049942.1 Escherichia phage JLK-2012	Count	1	1	2	1	1	1	1
	Ident	15	15	16	15	15	15	15
NC_022098.1 Pandoravirus salinus	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_031008.1 Staphylococcus phage SLPW	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	15	0
NC_061449.1 Erwinia phage pEa_SNUABM_17	Count	0	0	0	1	0	0	0
	Ident	0	0	0	15	0	0	0
NC_004156.2 Helicoverpa zea nudivirus 2	Count	0	1	1	1	1	0	0
	Ident	0	14	14	14	14	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	13	0
NC_006633.1 Cotesia congregata virus complete ge..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	12	0
NC_052978.1 Proteus phage Saba	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	11	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	0	1	8	1	0	7	0
	Ident	0	11	11	11	0	11	0
Novel mapping*	Count	38	281	970	335	59	650	9
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

Table 5.6 Viral genomes mapping to the Fowkes Pea 20 dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_041925.1 Proteus phage VB_PmiS-Isfahan	Count	240	3067	11647	4785	783	5029	3
	Ident	99	99	99	99	99	99	99
NC_003854.1 Pea enation mosaic virus satellite RNA	Count	13	13	8	15	13	15	8
	Ident	94	94	96	94	94	94	96
NC_003629.1 Pea enation mosaic virus-1	Count	2072	2520	3503	2776	2081	2643	1308
	Ident	94	94	94	94	94	94	94
NC_002766.1 Beet chlorosis virus	Count	1	1	1	1	1	1	1
	Ident	94	94	94	94	94	94	94
NC_003743.1 Turnip yellows virus	Count	21	40	75	47	23	46	13
	Ident	91	91	92	91	91	91	93
NC_003853.1 Pea enation mosaic virus-2	Count	730	725	548	734	707	667	461
	Ident	91	91	91	91	91	91	91
NC_003491.1 Beet mild yellowing virus	Count	2	2	2	2	2	2	2
	Ident	90	90	90	90	90	90	90
NC_016038.2 Brassica yellows virus isolate BrYV-..	Count	23	23	19	23	23	22	18
	Ident	90	90	88	90	90	89	88
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	4	72	294	115	23	132	1
	Ident	88	88	88	88	88	88	88
NC_055495.1 Faba bean polerovirus 1 strain 5253	Count	3	3	1	3	3	3	1
	Ident	86	86	80	86	86	86	80
NC_004756.1 Beet western yellows virus	Count	1	1	1	1	1	1	1
	Ident	82	82	82	82	82	82	82
NC_029993.1 Alfalfa enamovirus-1 isolate Manfredi	Count	10	10	7	10	10	10	7
	Ident	79	79	81	79	79	79	81
NC_043329.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	79	0	0	0
NC_009823.1 Hepatitis C virus genotype 2	Count	0	1	4	1	1	3	0
	Ident	0	66	56	66	66	56	0
NC_036582.1 Flamingopox virus FGPVKD09	Count	0	1	6	1	0	2	0
	Ident	0	36	64	36	0	36	0
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	1	2	1	3	1	1	1
	Ident	64	47	64	53	64	64	64
NC_001747.1 Potato leafroll virus	Count	0	1	10	5	0	9	0
	Ident	0	63	48	47	0	49	0
NC_022096.1 Pseudomonas phage PaBG	Count	0	0	1	0	0	0	0
	Ident	0	0	63	0	0	0	0
NC_008586.1 Ecotropis obliqua NPV	Count	0	1	1	1	0	1	0
	Ident	0	62	62	62	0	62	0
NC_002520.1 Amsacta moorei entomopoxvirus 'L'	Count	0	0	1	0	0	0	0
	Ident	0	0	59	0	0	0	0
NC_028112.1 Yellowstone lake phycodnavirus 1 DNA	Count	1	3	20	4	1	9	0
	Ident	47	47	58	58	47	47	0
NC_028250.1 Rosellinia necatrix partitivirus 6 C..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	58	0	0	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	0	5	23	8	1	13	0
	Ident	0	38	57	38	57	40	0
NC_043223.1 Senegalvirus SSV-A contig6 genomic s..	Count	0	0	1	0	0	0	0
	Ident	0	0	57	0	0	0	0
NC_008168.1 Choristoneura fumiferana granulovirus	Count	49	540	1993	781	132	798	0
	Ident	43	44	33	40	54	26	0
NC_034265.1 Tobacco virus 2	Count	3	41	103	44	5	33	0
	Ident	42	42	44	42	45	53	0
NC_004102.1 Hepatitis C virus genotype 1	Count	15	195	728	271	43	314	0
	Ident	26	26	53	26	26	34	0
NC_048049.1 Synechococcus phage S-T4	Count	13	154	639	240	40	274	0
	Ident	52	52	52	52	52	52	0

Table 5.6 Viral genomes mapping to the Fowkes Pea 20 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_029302.1 Piscine myocarditis-like virus isola..	Count	0	0	1	0	0	0	0
	Ident	0	0	52	0	0	0	0
NC_032255.1 Plodia interpunctella granulovirus i..	Count	0	1	1	0	0	2	0
	Ident	0	50	50	0	0	37	0
NC_006882.2 Prochlorococcus phage P-SSP7	Count	0	1	0	1	0	3	0
	Ident	0	49	0	49	0	42	0
NC_048171.1 Synechococcus phage S-B28	Count	0	5	4	6	1	1	0
	Ident	0	41	32	37	46	41	0
NC_013756.1 Marseillevirus marseillevirus strain..	Count	1	1	1	2	0	3	0
	Ident	39	39	39	39	0	46	0
NC_055710.1 Synechococcus phage S-CAM22 isolate ..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	46	0
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	0	4	10	4	0	5	0
	Ident	0	43	29	43	0	24	0
NC_031944.1 Synechococcus phage S-WAM1 isolate 0..	Count	0	0	1	0	0	0	0
	Ident	0	0	43	0	0	0	0
NC_006820.1 Synechococcus phage S-PM2	Count	0	0	1	0	0	0	0
	Ident	0	0	43	0	0	0	0
NC_005309.1 Canarypox virus	Count	0	0	1	0	0	0	0
	Ident	0	0	41	0	0	0	0
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	8	83	372	144	13	157	0
	Ident	28	28	41	36	28	28	0
NC_030230.1 Tokyovirus A1 DNA	Count	0	1	5	2	0	3	0
	Ident	0	32	23	32	0	40	0
NC_020864.1 Micromonas pusilla virus 12T genomic..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	40	0
NC_071036.1 Stenotrophomonas phage vB_SmaS-AXL_3	Count	0	0	1	0	0	0	0
	Ident	0	0	40	0	0	0	0
NC_022646.1 Clostera anastomosis granulovirus He..	Count	0	3	7	5	1	3	0
	Ident	0	38	38	38	38	38	0
NC_015282.1 Synechococcus phage S-SM1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	38	0
NC_025412.1 Melbournevirus isolate 1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	38	0	0	0
NC_043235.1 Paramecium bursaria Chlorella virus ..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	37	0	0	0
NC_028663.1 Cyanophage P-TIM40	Count	15	135	507	202	30	197	0
	Ident	27	31	30	31	27	36	0
NC_043352.1 Glyptapanteles indiensis bracovirus ..	Count	0	0	2	0	0	0	0
	Ident	0	0	36	0	0	0	0
NC_070848.1 Vibrio phage BUCT194	Count	0	1	0	1	0	0	0
	Ident	0	36	0	36	0	0	0
NC_001987.1 Ateline herpesvirus 3 complete genome	Count	0	1	0	1	1	0	0
	Ident	0	35	0	35	35	0	0
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	6	71	297	126	19	107	0
	Ident	27	27	35	27	27	27	0
NC_061449.1 Erwinia phage pEa_SNUABM_17	Count	0	0	2	1	0	1	0
	Ident	0	0	35	35	0	35	0
NC_048015.1 Cyanophage S-TIM4	Count	0	1	0	1	0	0	0
	Ident	0	35	0	35	0	0	0
NC_024697.1 Aureococcus anophagefferens virus is..	Count	0	2	3	4	1	1	0
	Ident	0	24	34	24	28	28	0
NC_002816.1 Cydia pomonella granulovirus	Count	0	1	2	2	1	0	0
	Ident	0	32	33	33	32	0	0
NC_071044.1 Bacillus phage vB_BanS_Nate	Count	1	12	53	20	1	20	0
	Ident	25	25	25	33	25	25	0
NC_048651.1 Aeribacillus phage AP45	Count	0	8	37	10	1	12	0
	Ident	0	28	25	31	24	26	0

Table 5.6 Viral genomes mapping to the Fowkes Pea 20 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_009827.1 Hepatitis C virus genotype 6	Count	0	0	1	0	0	0	0
	Ident	0	0	31	0	0	0	0
NC_015283.1 Prochlorococcus phage P-RSM4	Count	0	0	0	0	0	2	0
	Ident	0	0	0	0	0	31	0
NC_053004.1 Salmonella phage TS13	Count	29	388	1434	561	106	619	0
	Ident	29	28	28	28	27	28	0
NC_028095.1 Torulaspora delbrueckii dsRNA Mbarr-..	Count	0	1	3	1	1	1	0
	Ident	0	27	27	27	27	27	0
NC_002188.1 Fowlpox virus	Count	0	0	1	0	0	0	0
	Ident	0	0	26	0	0	0	0
NC_020867.1 Synechococcus phage S-RIP1 genomic s..	Count	0	0	1	0	0	0	0
	Ident	0	0	26	0	0	0	0
NC_037665.1 Pandoravirus macleodensis	Count	0	0	0	1	0	0	0
	Ident	0	0	0	25	0	0	0
NC_008929.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	25	0	0	0	0
NC_036594.1 Orpheovirus IHUMI-LCC2 genome assembly	Count	0	1	12	2	0	7	0
	Ident	0	25	25	25	0	25	0
NC_061448.1 Erwinia phage pEa_SNUABM_1	Count	0	0	1	0	0	2	0
	Ident	0	0	24	0	0	23	0
NC_003389.1 Swinepox virus	Count	0	0	1	0	0	0	0
	Ident	0	0	24	0	0	0	0
NC_047813.1 Staphylococcus phage Andhra	Count	0	8	8	6	1	5	0
	Ident	0	23	20	23	19	19	0
NC_015289.1 Synechococcus phage S-SSM5	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_014637.1 Cafeteria roenbergensis virus BV-PW1	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_008908.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	0	0	1	0	0	0	0
	Ident	0	0	22	0	0	0	0
NC_002687.1 Ectocarpus siliculosus virus 1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	22	0	0	0
NC_022098.1 Pandoravirus salinus	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	22	0
NC_007609.1 Dulcamara mottle virus	Count	0	2	2	2	0	0	0
	Ident	0	22	22	22	0	0	0
NC_049942.1 Escherichia phage JLK-2012	Count	0	0	1	0	0	2	0
	Ident	0	0	16	0	0	21	0
NC_029032.1 Phormidium phage MIS-PhV1A	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	0	0	2	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_019495.1 Cyprinid herpesvirus 2 strain ST-J1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	18	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	0	2	7	2	0	2	0
	Ident	0	16	17	16	0	18	0
NC_006656.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_002642.1 Yaba-like disease virus	Count	0	0	0	1	0	0	0
	Ident	0	0	0	17	0	0	0
NC_022597.1 Murrumbidgee virus isolate 934 segme..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	17	0
NC_048804.1 Stenotrophomonas phage Mendera	Count	1	1	1	0	0	0	0
	Ident	15	15	16	0	0	0	0
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	0	4	18	4	1	4	0
	Ident	0	15	16	15	14	14	0

Table 5.6 Viral genomes mapping to the Fowkes Pea 20 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_024502.1 Gentian ovary ring-spot virus genomi..	Count	0	1	0	1	0	0	0
	Ident	0	15	0	15	0	0	0
NC_055577.1 Physalis rugose mosaic virus isolate..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	14	0
NC_052978.1 Proteus phage Saba	Count	1	1	0	0	0	0	0
	Ident	13	13	0	0	0	0	0
Novel mapping*	Count	83	990	3786	1450	249	1589	1
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset.

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_003056.1 Soybean dwarf virus genomic RNA	Count	815	2434	16878	4247	381	5406	3
	Ident	97	97	97	97	97	97	97
NC_003629.1 Pea enation mosaic virus-1	Count	4015	3970	4090	4051	3959	4063	3741
	Ident	97	97	97	97	97	97	97
NC_003853.1 Pea enation mosaic virus-2	Count	262	253	259	263	259	264	242
	Ident	92	92	92	92	92	92	92
NC_003743.1 Turnip yellows virus	Count	2	2	2	2	2	2	2
	Ident	91	91	91	91	91	91	91
NC_040615.1 Eptesicus fuscus gammaherpesvirus	Count	0	0	17	3	0	5	0
	Ident	0	0	74	74	0	74	0
NC_043054.1 Bubaline alphaherpesvirus 1 strain b6	Count	66	196	1473	333	29	465	1
	Ident	73	73	73	73	73	73	73
NC_033775.1 Noumeavirus isolate NMV1	Count	0	0	2	0	0	1	0
	Ident	0	0	48	0	0	67	0
NC_029302.1 Piscine myocarditis-like virus isola..	Count	0	0	5	1	0	2	0
	Ident	0	0	65	29	0	67	0
NC_028112.1 Yellowstone lake phycodnavirus 1 DNA	Count	0	5	24	10	1	5	0
	Ident	0	47	61	66	47	31	0
NC_024382.1 Alcelaphine herpesvirus 2 isolate to..	Count	0	0	4	0	0	0	0
	Ident	0	0	62	0	0	0	0
NC_006560.1 Cercopithecine herpesvirus 2	Count	0	0	1	0	0	0	0
	Ident	0	0	56	0	0	0	0
NC_043352.1 Glyptapanteles indiensis bracovirus ..	Count	0	1	2	1	0	0	0
	Ident	0	56	52	56	0	0	0
NC_029692.1 Brazilian marseillevirus strain BH2014	Count	0	1	6	1	0	1	0
	Ident	0	35	56	35	0	35	0
NC_020855.1 Cyanophage P-RSM6 genomic sequence	Count	0	2	1	1	1	0	0
	Ident	0	36	20	52	20	0	0
NC_048049.1 Synechococcus phage S-T4	Count	52	207	1363	328	27	438	0
	Ident	51	51	28	26	51	51	0
NC_001782.1 Saccharomyces cerevisiae killer viru..	Count	0	0	1	0	0	0	0
	Ident	0	0	49	0	0	0	0
NC_008586.1 Ecotropis obliqua NPV	Count	0	0	1	0	0	2	0
	Ident	0	0	48	0	0	39	0
NC_009898.1 Paramecium bursaria Chlorella virus ..	Count	0	0	4	1	0	0	0
	Ident	0	0	46	38	0	0	0
NC_032111.1 BeAn 58058 virus	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	43	0
NC_022098.1 Pandoravirus salinus	Count	0	1	3	1	0	1	0
	Ident	0	42	27	42	0	27	0
NC_028094.1 Chrysochromulina ericina virus isola..	Count	2	8	72	12	3	29	0
	Ident	20	20	18	42	20	17	0
NC_032255.1 Plodia interpunctella granulovirus i..	Count	3	9	45	8	1	13	0
	Ident	41	41	41	27	27	38	0
NC_008912.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	41	0	0	0	0
NC_025412.1 Melbournevirus isolate 1	Count	1	3	11	5	0	2	0
	Ident	27	24	40	24	0	27	0
NC_015289.1 Synechococcus phage S-SSM5	Count	27	92	610	167	19	176	0
	Ident	16	40	25	40	40	16	0
NC_006820.1 Synechococcus phage S-PM2	Count	2	3	51	9	3	13	0
	Ident	21	21	30	39	21	21	0
NC_034265.1 Tobacco virus 2	Count	2	8	37	6	0	10	0
	Ident	36	38	33	39	0	33	0
NC_070848.1 Vibrio phage BUCT194	Count	0	0	1	0	0	0	0
	Ident	0	0	38	0	0	0	0

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_013756.1 Marseillevirus marseillevirus strain..	Count	0	1	6	3	0	2	0
	Ident	0	36	28	26	0	24	0
NC_008724.1 Acanthocystis turfacea Chlorella vir..	Count	1	2	14	4	0	5	0
	Ident	27	27	35	27	0	27	0
NC_006883.2 Prochlorococcus phage P-SSM2	Count	0	0	1	0	0	1	0
	Ident	0	0	34	0	0	34	0
NC_001747.1 Potato leafroll virus	Count	108	326	2306	554	54	707	0
	Ident	32	32	33	32	22	33	0
NC_021858.1 Pandoravirus dulcis	Count	0	0	8	0	0	3	0
	Ident	0	0	20	0	0	33	0
NC_031032.1 Bacillus phage Stitch	Count	0	1	5	1	0	1	0
	Ident	0	33	32	33	0	25	0
NC_009823.1 Hepatitis C virus genotype 2	Count	0	3	18	5	0	4	0
	Ident	0	31	32	30	0	27	0
NC_024709.1 Ball python nidovirus strain 07-53	Count	0	3	12	2	1	3	0
	Ident	0	32	32	32	32	32	0
NC_028478.1 Tomato brown rugose fruit virus isol..	Count	0	0	2	0	0	0	0
	Ident	0	0	32	0	0	0	0
NC_031944.1 Synechococcus phage S-WAM1 isolate 0..	Count	3	7	80	22	1	25	0
	Ident	32	32	23	32	32	32	0
NC_002816.1 Cydia pomonella granulovirus	Count	0	0	1	0	0	0	0
	Ident	0	0	31	0	0	0	0
NC_009758.1 Marine RNA virus JP-B	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_021536.1 Synechococcus phage S-IOM18 genomic ..	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_002520.1 Amsacta moorei entomopoxvirus 'L'	Count	0	0	1	0	0	0	0
	Ident	0	0	30	0	0	0	0
NC_019491.1 Cyprinid herpesvirus 1 strain NG-J1	Count	1	1	6	1	0	3	0
	Ident	30	30	30	15	0	30	0
NC_001422.1 Escherichia phage phiX174	Count	2	3	11	3	2	4	2
	Ident	29	29	26	29	29	29	29
NC_048171.1 Synechococcus phage S-B28	Count	0	1	5	2	0	1	0
	Ident	0	27	27	29	0	29	0
NC_015326.1 Lausannevirus	Count	1	1	10	1	0	2	0
	Ident	29	29	29	29	0	29	0
NC_061449.1 Erwinia phage pEa_SNUABM_17	Count	0	1	14	1	0	7	0
	Ident	0	16	25	16	0	29	0
NC_070962.1 Synechococcus phage S-SCSM1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	29	0	0	0
NC_004102.1 Hepatitis C virus genotype 1	Count	2	6	49	16	2	11	0
	Ident	19	20	29	18	20	20	0
NC_029993.1 Alfalfa enamovirus-1 isolate Manfredi	Count	57	56	55	58	57	57	54
	Ident	28	29	27	29	28	28	28
NC_005068.1 Cryptophlebia leucotreta granulovirus	Count	0	0	2	0	0	0	0
	Ident	0	0	28	0	0	0	0
NC_033436.1 Wuchan romanomermis nematode virus 2..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	28	0
NC_031922.1 Synechococcus phage S-CAM9 isolate 1..	Count	0	0	2	0	0	0	0
	Ident	0	0	28	0	0	0	0
NC_022646.1 Clostera anastomosis granulovirus He..	Count	1	2	10	1	0	1	0
	Ident	17	22	18	28	0	17	0
NC_041831.1 Campylobacter phage vB_CcoM-IBB_35 c..	Count	27	68	440	96	8	115	1
	Ident	18	26	27	26	18	25	18
NC_020490.2 Staphylococcus phage StB12	Count	0	0	1	0	0	0	0
	Ident	0	0	27	0	0	0	0
NC_021312.1 Phaeocystis globosa virus strain 16T	Count	0	0	2	0	0	0	0
	Ident	0	0	27	0	0	0	0

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_038425.1 Non-primate hepacivirus NZP1 polypro..	Count	3	9	45	16	0	14	0
	Ident	21	23	23	23	0	26	0
NC_038378.1 Cacao swollen shoot CD virus isolate..	Count	0	0	1	0	0	0	0
	Ident	0	0	26	0	0	0	0
NC_070761.1 Gordonia phage GMA2	Count	0	1	5	1	0	1	0
	Ident	0	26	26	26	0	26	0
NC_009127.1 Cyprinid herpesvirus 3	Count	0	0	3	0	0	1	0
	Ident	0	0	26	0	0	21	0
NC_043329.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	1	0	0	0	0
	Ident	0	0	26	0	0	0	0
NC_055365.1 Gordil virus isolate Dak ANBr 496d s..	Count	0	0	7	1	0	1	0
	Ident	0	0	25	25	0	25	0
NC_040536.1 Esparto virus isolate SRR3939042_Esp..	Count	1	3	8	3	0	4	0
	Ident	22	25	22	25	0	22	0
NC_047733.1 Synechococcus phage S-RIM8 isolate R..	Count	0	0	1	0	0	0	0
	Ident	0	0	25	0	0	0	0
NC_024697.1 Aureococcus anophagefferens virus is..	Count	0	0	1	0	0	1	0
	Ident	0	0	24	0	0	25	0
NC_011345.1 Agrotis ipsilon multiple nucleopolyh..	Count	0	0	3	0	0	1	0
	Ident	0	0	25	0	0	25	0
NC_020104.1 Acanthamoeba polyphaga moumouvirus	Count	0	0	7	1	1	2	0
	Ident	0	0	25	23	23	23	0
NC_026440.1 Pandoravirus inopinatum isolate KlaHel	Count	0	0	1	2	2	0	0
	Ident	0	0	25	22	22	0	0
NC_055230.1 Akhmeta virus isolate Akhmeta_2013-88	Count	0	1	4	1	0	2	0
	Ident	0	25	15	25	0	25	0
NC_037665.1 Pandoravirus macleodensis	Count	0	0	17	2	0	5	0
	Ident	0	0	21	23	0	25	0
NC_038828.1 Heterobasidion RNA virus 1 isolate H..	Count	0	0	4	1	0	1	0
	Ident	0	0	24	19	0	19	0
NC_008168.1 Choristoneura fumiferana granulovirus	Count	207	688	4628	1163	102	1438	0
	Ident	22	16	23	15	21	24	0
NC_034557.1 Imjin virus segment M glycoprotein g..	Count	0	0	1	0	0	0	0
	Ident	0	0	24	0	0	0	0
NC_001266.1 Rabbit fibroma virus	Count	0	0	1	0	0	1	0
	Ident	0	0	24	0	0	24	0
NC_014637.1 Cafeteria roenbergensis virus BV-PW1	Count	1	3	20	4	0	7	0
	Ident	23	23	23	23	0	23	0
NC_030230.1 Tokyovirus A1 DNA	Count	0	0	4	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_006151.1 Suid herpesvirus 1	Count	0	0	8	1	0	4	0
	Ident	0	0	23	23	0	23	0
NC_043307.1 Diolcogaster facetosa bracovirus seg..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_043508.1 Persea americana chrysovirus segment..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_006647.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_009827.1 Hepatitis C virus genotype 6	Count	0	0	7	3	0	6	0
	Ident	0	0	23	23	0	23	0
NC_013668.3 Anguillid herpesvirus 1	Count	1	2	6	2	0	1	0
	Ident	23	23	23	23	0	23	0
NC_015283.1 Prochlorococcus phage P-RSM4	Count	0	0	1	0	0	0	0
	Ident	0	0	23	0	0	0	0
NC_033774.1 Pepper chlorotic spot virus isolate ..	Count	46	129	768	213	19	254	0
	Ident	23	21	19	21	20	19	0
NC_031001.1 Gordonia phage Terapin	Count	0	0	2	0	0	2	0
	Ident	0	0	22	0	0	22	0

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_053004.1 Salmonella phage TS13	Count	399	1256	8156	2082	176	2576	1
	Ident	22	21	21	21	21	21	21
NC_026421.1 Equid herpesvirus 5 strain 2-141/67	Count	0	0	4	0	0	2	0
	Ident	0	0	22	0	0	22	0
NC_033829.1 Kallithea virus isolate DrosEU46_Kha..	Count	0	1	5	1	1	1	0
	Ident	0	11	21	11	11	21	0
NC_020859.1 Synechococcus phage S-RIM2 R1_1999	Count	0	0	2	0	0	0	0
	Ident	0	0	21	0	0	0	0
NC_024303.1 Bovine herpesvirus 6 isolate Pennsyl..	Count	0	0	2	0	0	1	0
	Ident	0	0	21	0	0	21	0
NC_071140.1 Escherichia phage ZCEC13	Count	23	53	460	96	14	132	0
	Ident	21	21	21	21	21	21	0
NC_055467.1 Cyclophragma undans nucleopolyhedrov..	Count	0	0	1	0	0	1	0
	Ident	0	0	18	0	0	21	0
NC_001493.2 Ictalurid herpesvirus 1 strain Aubur..	Count	0	0	3	0	0	0	0
	Ident	0	0	21	0	0	0	0
NC_048804.1 Stenotrophomonas phage Mendera	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_055166.1 Beluga whale alphaherpesvirus 1 stra..	Count	0	0	1	0	0	0	0
	Ident	0	0	20	0	0	0	0
NC_049942.1 Escherichia phage JLK-2012	Count	1	1	5	1	0	3	0
	Ident	15	15	20	15	0	15	0
NC_055231.1 Orthopoxvirus Abatino	Count	1	1	0	0	0	0	0
	Ident	20	20	0	0	0	0	0
NC_038332.1 Fowl adenovirus 6 strain CR119	Count	1	1	3	2	0	2	0
	Ident	20	20	20	20	0	20	0
NC_025257.1 Erinnyis ello granulovirus	Count	0	0	4	1	0	2	0
	Ident	0	0	14	14	0	20	0
NC_047760.1 Lactobacillus phage SA-C12	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	20	0
NC_002687.1 Ectocarpus siliculosus virus 1	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	19	0
NC_027925.1 Apis mellifera filamentous virus iso..	Count	5	14	125	29	3	42	0
	Ident	19	19	19	19	19	19	0
NC_022615.1 Dill cryptic virus 1 isolate IPP_hor..	Count	1	1	14	3	2	4	0
	Ident	19	19	19	19	19	19	0
NC_007346.1 Emiliania huxleyi virus 86	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_004010.1 Potato virus V	Count	0	0	1	0	0	0	0
	Ident	0	0	19	0	0	0	0
NC_047813.1 Staphylococcus phage Andhra	Count	0	0	9	1	0	2	0
	Ident	0	0	17	17	0	17	0
NC_061448.1 Erwinia phage pEa_SNUABM_1	Count	0	0	0	1	0	0	0
	Ident	0	0	0	17	0	0	0
NC_055410.1 Caimito virus isolate VP-488A segmen..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	17	0
NC_054922.1 Escherichia phage vB_EcoM_KAW1E185	Count	36	116	740	195	14	250	0
	Ident	17	14	15	15	17	16	0
NC_037666.1 Pandoravirus neocaledonia	Count	0	1	8	2	0	5	0
	Ident	0	16	17	16	0	16	0
NC_020867.1 Synechococcus phage S-RIP1 genomic s..	Count	0	1	5	1	0	0	0
	Ident	0	16	17	16	0	0	0
NC_001798.2 Human herpesvirus 2 strain HG52	Count	0	0	1	0	0	0	0
	Ident	0	0	17	0	0	0	0
NC_028091.1 Ostreococcus lucimarinus virus 2 iso..	Count	0	0	0	1	0	0	0
	Ident	0	0	0	17	0	0	0
NC_007609.1 Dulcamara mottle virus	Count	0	0	1	0	0	1	0
	Ident	0	0	13	0	0	17	0

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_049835.1 Klebsiella phage vB_KpnS_Domnhall	Count	0	0	5	0	0	1	0
	Ident	0	0	17	0	0	17	0
NC_036600.1 Rosellinia necatrix partitivirus 8 g..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_013221.1 Phytophthora infestans RNA virus 1 R..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	16	0
NC_038553.1 Heterosigma akashiwo virus 01 isolat..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_038838.1 Crimson clover cryptic virus 2 isola..	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_037052.1 Pepper enamovirus isolate R1 ORF0	Count	1	1	1	1	1	1	1
	Ident	16	16	16	16	16	16	16
NC_016447.1 Aotine herpesvirus 1 strain S34E	Count	0	0	1	0	0	0	0
	Ident	0	0	16	0	0	0	0
NC_044937.1 Paramecium bursaria Chlorella virus ..	Count	0	0	2	1	0	0	0
	Ident	0	0	15	15	0	0	0
NC_028250.1 Rosellinia necatrix partitivirus 6 C..	Count	0	0	0	0	0	1	0
	Ident	0	0	0	0	0	15	0
NC_036582.1 Flamingopox virus FGPVKD09	Count	0	1	0	0	0	0	0
	Ident	0	15	0	0	0	0	0
NC_054919.1 Escherichia phage vB_EcoM_G4507	Count	5	24	157	39	7	56	0
	Ident	15	13	14	13	13	14	0
NC_040681.1 Bufonid herpesvirus 1 strain FO1_2015	Count	0	0	1	0	0	0	0
	Ident	0	0	15	0	0	0	0
NC_048651.1 Aeribacillus phage AP45	Count	63	231	1662	387	33	509	0
	Ident	14	15	14	14	13	14	0
NC_003038.1 Invertebrate iridescent virus 6	Count	0	0	1	0	0	0	0
	Ident	0	0	14	0	0	0	0
NC_011421.1 Bacillus phage SPO1	Count	0	0	4	2	0	2	0
	Ident	0	0	14	14	0	14	0
NC_055142.1 Lymphocryptovirus Macaca/pfe-lcl-E3	Count	0	3	19	5	1	16	0
	Ident	0	14	14	14	14	13	0
NC_026242.1 Tipula oleracea nudivirus isolate 35	Count	1	0	2	0	0	0	0
	Ident	14	0	13	0	0	0	0
NC_004067.1 Pepino mosaic virus	Count	0	1	4	3	1	1	0
	Ident	0	14	11	14	14	14	0
NC_049948.1 Escherichia phage Lambda_ev017 genom..	Count	11	33	196	56	9	76	0
	Ident	13	13	14	13	13	13	0
NC_024111.1 Jingmen Tick Virus isolate SY84 segm..	Count	0	0	3	1	0	3	0
	Ident	0	0	14	14	0	14	0
NC_043569.1 Iaco virus strain BeAn314206 nucleoc..	Count	1	1	2	2	0	0	0
	Ident	13	13	13	13	0	0	0
NC_049372.1 Roseobacter phage RD-1410W1-01	Count	0	1	13	4	0	6	0
	Ident	0	13	13	13	0	12	0
NC_052978.1 Proteus phage Saba	Count	4	17	113	21	3	35	0
	Ident	12	13	12	12	12	12	0
NC_024384.1 Listeria phage LP-030-3	Count	0	0	1	0	0	1	0
	Ident	0	0	13	0	0	13	0
NC_035470.1 Abisko virus isolate Abisko-6	Count	0	0	2	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_007993.1 Campoletis sonorensis ichnovirus seg..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_043176.1 Oxbow virus strain Ng1453 glycoprote..	Count	0	5	23	5	0	5	0
	Ident	0	11	11	11	0	12	0
NC_048798.1 Klebsiella phage Marfa	Count	12	47	322	73	13	95	0
	Ident	12	12	12	12	12	12	0
NC_006633.1 Cotesia congregata virus complete ge..	Count	0	0	1	0	0	1	0
	Ident	0	0	12	0	0	12	0

Table 5.7 Viral genomes mapping to the Fowkes Pea 15 dataset (cont.)

Viral Mapping	Stat	MMS2	HMM3	DVF	MS	GA	PR	UN
NC_030953.1 Shigella phage SHFML-11	Count	0	0	5	1	0	0	0
	Ident	0	0	12	11	0	0	0
NC_043313.1 Diolcogaster facetosa bracovirus clo..	Count	0	0	1	0	0	0	0
	Ident	0	0	12	0	0	0	0
NC_049392.1 Escherichia phage ESSI2_ev239 genome..	Count	0	1	9	1	0	5	0
	Ident	0	12	11	12	0	12	0
NC_008898.1 Glypta fumiferanae ichnovirus segmen..	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
NC_010489.1 Tomato zonate spot virus segment S	Count	0	0	0	1	0	0	0
	Ident	0	0	0	11	0	0	0
NC_038882.1 Hepatitis C virus strain H77 pCV-H77..	Count	0	0	1	0	0	0	0
	Ident	0	0	11	0	0	0	0
Novel mapping*	Count	849	2556	17123	4294	408	5418	0
	Ident	0	0	0	0	0	0	0

Count: Total number of mapped reads that are labelled as viral by each tool. Ident: Median percentage identity of labelled reads. MMS2: MMseqs2, HMM3: HMMER3, DVF: DeepVirFinder, MS: Mash Screen, PR: Path Racer, UN: Unanimously labelled reads.
*Novel mapping includes any read labelled as viral that was not mapped to a viral genome by Bowtie2.

5.4. Conclusions

In this chapter, we compared and contrasted the outputs of a diverse set of virus detection software in labelling reads as being of viral or non-viral origin. We compared this behaviour on multiple levels, investigating the overall distribution in the amount of corroboration between tools, the overlaps between their outputs, the putative viral reads in common between subsets of tools, and the specific viral genomes that these reads map onto.

In all datasets, there was some measure of disagreement between virus detection software. The extent of this varied between datasets, where the single-sample Pea coinfection dataset showed a highly corroborated, near-unanimous, core set of viral reads that were generated from multiple isolates of Turnip Yellows Virus. On the other hand, the Fowkes Pea bulk sequencing datasets showed much disagreement between tools, even on genomes that had previously been confirmed to be present by RT-PCR (Soybean dwarf virus in the Fowkes Pea 15 datset). Overall, though, high-identity viral genomes with biological rationale had significant numbers of reads from all tools, even in the presence of disagreement between which exact reads this included. This shows that all approaches are robust in detecting known viruses, even when their methodologies give differences in output. The largest area that the disagreement between tools manifested is in the case of low-identity and non-mapping reads. Most datasets showed many novel mappings, especially by DeepVirFinder, the only tool to use a non-homology approach to viral genome detection. The nature of these divergent reads was not explored in this chapter. Indeed, some of these reads may belong to genomes at the limits of detection of all tools, such as where there is a combination of low read depth, genetic divergence, and/or sparse taxonomy, as was characterised in Chapter 4. Differentiating between false positives and genuine divergent sequences continues to be a significant challenge. The utility of including a non homology-based software, in this case DeepVirFinder, has been seen in its ability to corroborate the output of homology-based tools that is independent of reference databases. Genomes that had a similar number of DeepVirFinder hits to homology-based hits, especially when many individual reads were highly corroborated, were those that one would most expect to find in a plant virome.

In conclusion, methodologically diverse virus detection software detect different properties of viral genomes within metagenomic datasets, manifesting as differences in putative viral reads. The biggest difference was between homology and non-homology, but these approaches still showed an overall similarity in the known genomes they detected. By combining homology and non-homology software, one is able to reach more robust conclusions than either alone. Instead of relying on single tools, a multi-approach toolbox acts as viromic sieve with fewer holes. The continued development of new approaches continues to close these holes, shrinking the pool of viral dark matter.

Chapter 6. Discussion

6.1. Summary of work

In this study, we investigated the limits of current approaches in the detection of viral genomes in high-throughput plant metagenomic sequencing. Our aims at the beginning of the study were as follows:

1. Create bespoke software capable of calculating a numerical distance between viral genomes that is accurate to highly divergent sequences.
2. Quantify the factors influencing the limits of detection for current viral detection software
3. Carry out an in-depth qualitative analysis of the outputs of viral detection software on previously seen and novel sequencing datasets.

We believe that we have achieved the first two of these aims, and partially achieved the final.

In Chapter 3, we presented mottle, a novel approach for the calculation of substitution distance between divergent sequences. This method used short read mapping to identify prospective homologous regions, and gradient descent to filter out non-homologous hits. We additionally developed a bespoke mapper, mottle-map, that used the Fast Fourier Transform to find homologous regions even at high divergence. We tested this new tool against current approaches in calculating substitution distance, using two artificial genome benchmarks where substitution distance was known, and a benchmark that utilised full viral genomes but only compared taxonomic-level information. We found that the mottle-map was able to successfully calculate and accurate distance up to 0.66 substitutions per base pair, further than any other tested software, and successfully differentiate between genomes up to the Order/Class level, matching the performance of the most stable tool, Mash. This fulfilled the first of our aims, calculating a numerical distance between divergent viral genomes.

The second of our aims, quantifying the factors influencing limits of viral detection, was the focus of Chapter 4. We tested three major limiting factors - divergence of references, read depth, and reference database size. We found that all assembly-based tools were able to accurately detect viral reads up to Class level when there were nine reference genomes, but only up to the Order level when the number of reference genomes was reduced to one. This generally corresponded to a minimum distance of 2-3 substitutions per base pair, even further than the abilities of mottle-map to accurately quantify. All assembly-based tools, though, showed a degradation of performance at low read depths, especially when coupled with higher divergence.

Assembly-free approaches, on the other hand, generally showed the oppositive, performing well even to 1 times read depth but struggling to detect genomes past the Species level. An exception was seen with a homology-model graph-based tool, PathRacer, which behaved similarly to assembly-based tools. A trade-off between performance at high divergence and low read numbers was seen across approaches. Reference database size had a smaller effect for all approaches, but generally compounded scenarios where tools already struggled. This lead us to conclude in this chapter that a mixed approach would be needed when analysing any metagenomic dataset, as it cannot be known in advance whether a viral genome is present that is highly divergent or at low depth. The scenario of intersecting limitations, where there is a low read depth and a high divergence, was not covered by any approach, even when considering a combination of tools. Additionally, we were able to calculate optimal thresholds for each tool for the detection of viral reads at the maximum number of scenarios.

Chapter 5 centred on the qualitative analysis of benchmarked software on plant virome sequencing. We compared the outputs of these tools on six datasets from three separate studies, where half of the datasets had been previously been analysed and published, and half were novel. We found that there was a large amount of disagreement between approaches to whether specific reads were of viral origin, but a consensus as to which genomes were present. This extended to the seen datasets, where viral genomes that had previously been detected and subsequently confirmed by RT-PCR were detected by all tools. DeepVirFinder, the only non-homology approach tested, showed a great utility in acting as an additional confirmation of viral genomes when combined with a homology-based approach. There was a large fraction of unmapped putative viral reads in many of the datasets. We were not able to investigate the nature of these novel reads within the time limits of this thesis study. Questions such as whether these reads were able to form virus-like contigs, and whether there were hallmarks or virus/viroid genomes, such as animo acid conservation or evidence of stable RNA secondary structures, remain unanswered. We therefore only consider the final aim to have been partially fulfilled.

6.2. Limitations and wider context

While the effect was limited compared to other factors, reference database sparsity was seen to affect the ability of tools to tolerate the highest divergences. The Refseq genomics database (O'Leary et al., 2016) used for sequencing data analysis in Chapter 5 is a subset of the nt/nr used by BLAST and related software for homology search (National Library of Medicine, 2023), and does not cover all viral strains or putative assemblies. This may have contributed to the large 'novel mapping' fractions in its results, especially as many of these were produced by the only non-homology tool DeepVirFinder, which does not rely on a reference database. Addition of a comprehensive viral genome database to a standard reference database, such as (Goodacre et al., 2018) could increase sensitivity of homology-based tools. This may risk an increase of false positives, especially when including many closely related sequences. The creation of bespoke datasets, that include only representative sequences at the limits of detection, would be one way to further optimise viral detection. Databases have been created that only contain divergent

sequences, such as that of UniClust (Mirdita et al., 2017), which clustered protein sequences to 90%, 50%, or 30% identity, and created databases of cluster representatives for each set. A similar methodology was used to create a database of representative protein fragments generated from metagenomic datasets (Steinegger and Söding, 2018). More recently, this was also done for protein structures predicted by the machine learning software AlphaFold (Barrio-Hernandez et al., 2023). These *de novo* protein reference datasets have shown great promise for the discovery of previously unknown viral genomes (Nayfach et al., 2021), but still present their own limits, in terms of the correct identification of protein-coding regions, which would be lacking in viroid genomes, and the continued reliance on homology search.

Being based on a Convolutional Neural Network (CNN) machine learning approach allows DeepVirFinder to discover putative viral genomes that are too divergent for homology-based approaches. In our analyses, we had assumed that its ability in detecting viral genomes is completely independent of reference databases. It has been suggested, though, that the genomes used within training data can limit the types of viruses CNN approaches can detect (Sukhorukov et al., 2022). Truly unknown viral sequences may therefore still be missed. Other machine learning approaches have been developed for the purpose of viral genome detection and annotation, including Gradient Boosted Decision Trees (Ren et al., 2017), Recurrent Neural Networks (Liu et al., 2022), and Large Language Models (Flamholz et al., 2023), but these are likely also limited by their training data. Approaches that rely neither on homology nor non machine learning, then, may be an attractive addition to a viromics toolbox. One such category of approaches is the detection of contigs that have no known homology to reference sequences, but nevertheless show the hallmarks of biological sequences. These approaches may look for the presence of selective pressures within viral metagenomes (Ye and Tang, 2009) or properties of non-coding regions (Tobar-Tosse et al., 2013).

The focus of this study was placed on metagenomic techniques for virus detections, but the utilisation of computational approaches for virus detection extends beyond this. Notably, the use of biosensors and remote sensing for the screening of plants for live infection (Sellappan et al., 2022). The application of machine learning to these datasets shows great promise for non-invasively profiling plants for viral infection before the development of visible symptoms (Peng et al., 2022). While live sensors make a promising addition to the computational toolbox, they are currently limited to the targeted detection of specific viruses, and their applicability to asymptomatic persistent infections, such as those known to occur by *Partitiviridae*, *Endornavirus* (Roossinck, 2010), and *Tobamovirus* (Ilyas et al., 2022). Further development in this field would complement, but not replace, the growing importance of metagenomic techniques.

6.3. Conclusions

The use of metagenomic techniques for the computational detection of plant viruses and viroids is a relatively new and quickly developing field. Determining the limits of current approaches

allows us to objectively monitor new approaches, and direct the focus of future work. Just within the course of this thesis study, the prominence of machine learning and long-read sequencing has shifted from limited adoption to gaining widespread usage. In order to gain the most out of these technologies, we must not only push the development of new tools, but also the development of new benchmarks that are able to inform us of their limitations.

Appendix A. Full Conda Enviroment

Full Conda Enviroment.

```
name: bio
channels:
  - cyclus
  - pytorch
  - fastchan
  - bioconda
  - conda-forge
  - defaults
  - r
  - ostrokach
  - kennethshang
  - cctbx202105
dependencies:
  - _libgcc_mutex=0.1=conda_forge
  - _openmp_mutex=4.5=1_gnu
  - _r-mutex=1.0.1=anacondar_1
  - _sysroot_linux-64_curr_repodata_hack=3=h5bd9786_13
  - _tflow_select=2.3.0=mkl
  - abseil-cpp=20210324.2=h9c3ff4c_0
  - adjusttext=0.7.3.1=py_1
  - aiohttp=3.8.1=py39h3811e60_0
  - aiosignal=1.2.0=pyhd8ed1ab_0
  - alabaster=0.7.12=py_0
  - alsa-lib=1.2.3=h516909a_0
  - altair=5.0.1=pyhd8ed1ab_1
  - amply=0.1.4=py_0
  - anyio=3.4.0=py39hf3d152e_0
  - appdirs=1.4.4=pyh9f0ad1d_0
  - argon2-cffi=21.1.0=py39h3811e60_2
  - argtable2=2.13=h14c3975_1001
  - arrow=1.2.1=pyhd8ed1ab_0
  - art=2016.06.05=he1d7d6f_6
  - asciitree=0.3.3=py_2
  - astor=0.8.1=pyh9f0ad1d_0
  - astunparse=1.6.3=pyhd8ed1ab_0
  - async-timeout=4.0.1=pyhd8ed1ab_0
  - async_generator=1.10=py_0
  - atk-1.0=2.36.0=h3371d22_4
  - attrs=21.2.0=pyhd8ed1ab_0
  - babel=2.9.1=pyh44b312d_0
  - backcall=0.2.0=pyh9f0ad1d_0
  - backports=1.0=py_2
  - backports.functools_lru_cache=1.6.4=pyhd8ed1ab_0
  - bamtools=2.5.1=h9a82719_9
  - bandage=0.8.1=hc9558a2_2
  - bash=5.0.018=h0a1914f_0
  - bbmap=38.93=he522d1c_0
  - bcbio-gff=0.7.0=pyh7cba7a3_0
  - bcftools=1.14=h88f3f91_0
```

```
- bcrypt=4.0.1=py39h9fdd4d6_1
- beautifulsoup4=4.11.2=pyha770c72_0
- bedops=2.4.39=h7d875b9_1
- bedtools=2.30.0=h7d7f7ad_2
- bifrost=1.0.6=h2e03b76_0
- binaryornot=0.4.4=py_1
- binutils=2.36.1=hdd6e379_2
- binutils_impl_linux-64=2.36.1=h193b22a_2
- binutils_linux-64=2.36=hf3e587d_1
- bioawk=1.0=h5bf99c6_6
- bioconductor-annotate=1.72.0=r41hdfd78af_0
- bioconductor-annotationdbi=1.56.2=r41hdfd78af_0
- bioconductor-apeglm=1.16.0=r41hc247a5b_2
- bioconductor-biobase=2.54.0=r41hc0cfd56_2
- bioconductor-biocfilecache=2.2.0=r41hdfd78af_0
- bioconductor-biocgenerics=0.40.0=r41hdfd78af_0
- bioconductor-biocio=1.4.0=r41hdfd78af_0
- bioconductor-biocparallel=1.28.3=r41hc247a5b_1
- bioconductor-biomart=2.50.0=r41hdfd78af_0
- bioconductor-biostrings=2.62.0=r41hc0cfd56_2
- bioconductor-ctc=1.68.0=r41hdfd78af_0
- bioconductor-delayedarray=0.20.0=r41hc0cfd56_2
- bioconductor-deseq2=1.34.0=r41hc247a5b_3
- bioconductor-dexseq=1.40.0=r41hdfd78af_0
- bioconductor-edger=3.36.0=r41hc247a5b_2
- bioconductor-genefilter=1.76.0=r41h38f54d8_2
- bioconductor-genelendatabase=1.30.0=r41hdfd78af_1
- bioconductor-geneplotter=1.72.0=r41hdfd78af_0
- bioconductor-genomeinfodb=1.30.1=r41hdfd78af_0
- bioconductor-genomeinfodbdata=1.2.7=r41hdfd78af_2
- bioconductor-genomicalignments=1.30.0=r41hc0cfd56_2
- bioconductor-genomicfeatures=1.46.1=r41hdfd78af_0
- bioconductor-genomicranges=1.46.1=r41hc0cfd56_1
- bioconductor-go.db=3.14.0=r41hdfd78af_1
- bioconductor-goseq=1.46.0=r41hdfd78af_0
- bioconductor-iranges=2.28.0=r41hc0cfd56_2
- bioconductor-keggrest=1.34.0=r41hdfd78af_0
- bioconductor-limma=3.50.3=r41hc0cfd56_0
- bioconductor-matrixgenerics=1.6.0=r41hdfd78af_0
- bioconductor-qvalue=2.26.0=r41hdfd78af_0
- bioconductor-rhtslib=1.26.0=r41hc0cfd56_2
- bioconductor-rsamtools=2.10.0=r41hc247a5b_2
- bioconductor-rtracklayer=1.54.0=r41h171f361_4
- bioconductor-s4vectors=0.32.4=r41hc0cfd56_0
- bioconductor-summarizedexperiment=1.24.0=r41hdfd78af_0
- bioconductor-xvector=0.34.0=r41hc0cfd56_2
- bioconductor-zlibbioc=1.40.0=r41hc0cfd56_2
- bioconvert=0.4.3=py_0
- biom-format=2.1.14=py39h72bdee0_2
- biopython=1.78=py39h3811e60_2
- blas=1.1=openblas
- blast=2.12.0=pl5262h3289130_0
- bleach=4.1.0=pyhd8ed1ab_0
- blinker=1.4=py_1
- bokeh=2.3.3=py39hf3d152e_0
- boost-cpp=1.74.0=h312852a_4
- bowtie=1.3.1=py39h176da8b_0
- bowtie2=2.4.5=py39hd2f7db1_3
- brotli=1.0.9=h7f98852_6
- brotli-bin=1.0.9=h7f98852_6
- brotlipy=0.7.0=py39h3811e60_1003
- bwa-mem2=2.2.1=h9a82719_1
```

- bwidget=1.9.14=ha770c72_1
- bx-python=0.8.12=py39h5d76eff_0
- bzip2=1.0.8=h7f98852_4
- c-ares=1.18.1=h7f98852_0
- c-compiler=1.3.0=h7f98852_0
- ca-certificates=2023.08.22=h06a4308_0
- cachecontrol=0.12.10=pyhd8ed1ab_0
- cached-property=1.5.2=hd8ed1ab_1
- cached_property=1.5.2=pyha770c72_1
- cachetools=4.2.4=pyhd8ed1ab_0
- cairo=1.16.0=h6cf1ce9_1008
- canu=2.2=ha47f30e_0
- capnproto=0.6.1=hfc679d8_1
- catboost=1.0.3=py39hf3d152e_1
- cattrs=22.2.0=pyhd8ed1ab_0
- cd-hit=4.8.1=h2e03b76_5
- certifi=2023.7.22=pyhd8ed1ab_0
- cffi=1.15.1=py39he91dace_0
- chardet=4.0.0=py39hf3d152e_2
- charset-normalizer=2.0.8=pyhd8ed1ab_0
- chrpath=0.16=h7f98852_1002
- click=8.0.3=py39hf3d152e_1
- cliquesnv=2.0.3=hdfd78af_0
- cloudpickle=2.0.0=pyhd8ed1ab_0
- cmake=3.21.3=h8897547_0
- cmseq=1.0.4=pyhb7b1952_0
- coincbc=2.10.5=hcee13e7_1
- colorama=0.4.4=pyh9f0ad1d_0
- colorcet=2.0.6=pyhd8ed1ab_0
- colorlog=6.6.0=py39hf3d152e_0
- commonmark=0.9.1=py_0
- compilers=1.3.0=ha770c72_0
- conda=23.1.0=py39hf3d152e_0
- conda-build=3.23.3=py39hf3d152e_0
- conda-package-handling=1.7.3=py39h3811e60_1
- configargparse=1.5.3=pyhd8ed1ab_0
- cookiecutter=1.7.0=py_0
- coolbox=0.3.8=pyhdfd78af_0
- cooler=0.8.11=pyh5e36f6f_1
- coreutils=8.32=h7b6447c_0
- crossmap=0.6.0=pyhb7b1952_0
- cryptography=35.0.0=py39h95dcef6_2
- cudatoolkit=11.5.0=h36ae40a_9
- curl=7.87.0=h5eee18b_0
- cuttlefish=1.0.0=h2e03b76_1
- cxx-compiler=1.3.0=h4bd325d_0
- cycler=0.11.0=pyhd8ed1ab_0
- cython=0.29.24=py39he80948d_1
- cytoolz=0.11.2=py39h3811e60_1
- dashing=0.4.0=h735f0e5_3
- dask=2021.11.2=pyhd8ed1ab_0
- dask-core=2021.11.2=pyhd8ed1ab_0
- dataclasses=0.8=pyhc8e2a94_3
- datrie=0.8.2=py39h3811e60_3
- dbus=1.13.18=hb2f20db_0
- dcor=0.5.3=pyhd8ed1ab_0
- debugpy=1.5.1=py39he80948d_0
- decorator=5.1.0=pyhd8ed1ab_0
- deeptools=3.5.1=py_0
- deeptoolsintervals=0.1.9=py39h38f01e4_3
- defusedxml=0.7.1=pyhd8ed1ab_0
- dendropy=4.5.2=pyh3252c3a_0

```
- deprecated=1.2.14=pyh1a96a4e_0
- diamond=2.1.6=h5b5514e_0
- dill=0.3.4=pyhd8ed1ab_0
- disco=1.2=h2e03b76_4
- distributed=2021.11.2=py39hf3d152e_0
- dm-tree=0.1.6=py39hde0f152_1
- dna_features_viewer=3.1.0=pyh5e36f6f_0
- docutils=0.17.1=py39hf3d152e_1
- dotnet=7.0.102=ha770c72_0
- dotnet-aspnetcore=7.0.2=h41ceda8_0
- dotnet-runtime=7.0.2=h41ceda8_0
- dotnet-sdk=7.0.102=h41ceda8_0
- dsrc=2015.06.04=hc90279e_3
- easydev=0.12.0=pyh6c4a22f_0
- emboss=6.6.0=h440b012_4
- entrez-direct=16.2=he881be0_0
- entrypoints=0.3=py39hde42818_1002
- et_xmlfile=1.1.0=py39h06a4308_0
- ete3=3.1.2=pyh9f0ad1d_0
- exceptiongroup=1.1.0=pyhd8ed1ab_0
- expat=2.4.1=h9c3ff4c_0
- fabric=3.2.2=pyhd8ed1ab_0
- faiss=1.7.1=py39cuda112h5ca99f2_1_cuda
- fasteners=0.16.3=pyhd3eb1b0_0
- fastp=0.23.2=h79da9fb_0
- fasttree=2.1.11=hec16e2b_1
- fastx_toolkit=0.0.14=he1b5a44_8
- fbpca=1.0=py_0
- fftw=3.3.10=nompi_h77c792f_102
- filelock=3.9.0=pyhd8ed1ab_0
- fire=0.4.0=pyh44b312d_0
- fmt=9.1.0=h924138e_0
- font-ttf-dejavu-sans-mono=2.37=hab24e00_0
- font-ttf-inconsolata=3.000=h77eed37_0
- font-ttf-source-code-pro=2.038=h77eed37_0
- font-ttf-ubuntu=0.83=hab24e00_0
- fontconfig=2.13.1=hba837de_1005
- fonts-conda-ecosystem=1=0
- fonts-conda-forge=1=0
- fonttools=4.28.2=py39h3811e60_0
- fortran-compiler=1.3.0=h1990efc_0
- freetype=2.11.0=h70c0345_0
- fribidi=1.0.10=h36c2ea0_0
- frozendict=2.3.4=py39hb9d737c_0
- frozenlist=1.2.0=py39h3811e60_1
- fsspec=2021.11.1=pyhd8ed1ab_0
- future=0.18.2=py39hf3d152e_4
- gast=0.4.0=pyh9f0ad1d_0
- gatk4=4.2.3.0=hdfd78af_1
- gawk=5.1.0=h7f98852_0
- gcc=9.4.0=h192d537_1
- gcc_impl_linux-64=9.4.0=h03d3576_11
- gcc_linux-64=9.4.0=h391b98a_1
- gdk-pixbuf=2.42.6=h04a7f16_0
- genericrepeatfinder=1.0=h7d875b9_1
- gettext=0.21.1=h27087fc_0
- gfaffix=0.1.2.4=h779adbc_0
- gfastats=1.3.6=hdcf5f25_3
- gfatools=0.5=h5bf99c6_1
- gfortran=9.4.0=h2018a41_1
- gfortran_impl_linux-64=9.4.0=h0003116_11
- gfortran_linux-64=9.4.0=hf0ab688_1
```

- giflib=5.2.1=h36c2ea0_2
- git=2.34.1=pl5321hc30692c_0
- gitdb=4.0.9=pyhd8ed1ab_0
- gitpython=3.1.26=pyhd8ed1ab_0
- glib=2.74.1=h6239696_0
- glib-tools=2.74.1=h6239696_0
- glob2=0.7=py_0
- gmp=6.2.1=h58526e2_0
- gmpy2=2.1.0rc1=py39h78fa15d_0
- gnupg=2.3.3=h7853c96_0
- gnuplot=5.4.1=hec6539f_2
- go=1.10.3=hfc679d8_3
- go-core=1.10.3=h26a2512_3
- go_linux-64=1.10.3=h0f5337a_3
- google-auth=2.3.3=pyh6c4a22f_0
- google-auth-oauthlib=0.4.6=pyhd8ed1ab_0
- google-pasta=0.2.0=pyh8c360ce_0
- graphaligner=1.0.13=he1c1bb9_0
- graphite2=1.3.14=h23475e2_0
- grep=3.4=h9d02d08_1
- grpc-cpp=1.39.1=h850795e_1
- gsl=2.6=he838d99_2
- gst-plugins-base=1.18.5=hf529b03_0
- gstreamer=1.18.5=h76c114f_0
- gtk2=2.24.33=h539f30e_1
- gxx=9.4.0=h192d537_1
- gxx_impl_linux-64=9.4.0=h03d3576_11
- gxx_linux-64=9.4.0=h0316aca_1
- h5py=3.1.0=nompi_py39h25020de_100
- harfbuzz=3.0.0=h83ec7ef_1
- hdf5=1.10.6=nompi_h6a2412b_1114
- hdmedians=0.14.2=py39hce5d2b2_1
- heapdict=1.0.1=py_0
- hhsuite=3.3.0=py39pl5262h8f06ca0_2
- hifiasm_meta=hamtv0.2=h2e03b76_0
- hisat2=2.2.1=h1b792b2_3
- hmmer=3.3.2=h1b792b2_1
- hmmer2=2.3.2=h779adbc_6
- html5lib=1.1=pyh9f0ad1d_0
- htsbox=r346=h5bf99c6_1
- htslib=1.14=h9093b5e_0
- hyperlib=0.0.6=py39hf939315_0
- icu=68.2=h9c3ff4c_0
- idna=3.3=pyhd3eb1b0_0
- imagesize=1.3.0=pyhd8ed1ab_0
- importlib-metadata=4.8.2=py39hf3d152e_0
- importlib_resources=5.4.0=pyhd8ed1ab_0
- infernal=1.1.4=h779adbc_0
- iniconfig=1.1.1=pyh9f0ad1d_0
- insilicoseq=1.5.4=pyh5e36f6f_0
- intervaltree=3.1.0=pyhd3eb1b0_0
- invoke=2.2.0=pyhd8ed1ab_0
- ipykernel=6.5.1=py39hef51801_0
- ipython=7.30.0=py39hf3d152e_0
- ipython_genutils=0.2.0=py_1
- ipywidgets=7.6.5=pyhd8ed1ab_0
- iqtree=2.2.0.3=hb97b32f_1
- isa-l=2.30.0=ha770c72_4
- itsdangerous=2.1.2=pyhd8ed1ab_0
- java-jdk=8.45.14=0
- java-jre=8.45.14=0
- jax=0.2.25=pyhd8ed1ab_0

- jaxlib=0.1.73=py39hde0f152_2
- jbig=2.1=h7f98852_2003
- jedi=0.18.1=py39hf3d152e_0
- jellyfish=2.2.6=0
- jemalloc=5.2.1=h9c3ff4c_6
- jinja2=3.0.3=pyhd8ed1ab_0
- jinja2-time=0.2.0=py_2
- joblib=1.1.0=pyhd8ed1ab_0
- joypy=0.2.4=pyhd3deb0d_0
- jpeg=9d=h36c2ea0_0
- jq=1.6=h36c2ea0_1000
- json5=0.9.6=pyhd3eb1b0_0
- jsonschema=4.2.1=pyhd8ed1ab_0
- jupyter=1.0.0=py39hf3d152e_7
- jupyter_client=7.1.0=pyhd8ed1ab_0
- jupyter_console=6.4.0=pyhd8ed1ab_0
- jupyter_core=4.9.1=py39hf3d152e_1
- jupyter_server=1.12.1=pyhd8ed1ab_0
- jupyterlab=3.2.4=pyhd8ed1ab_0
- jupyterlab_pygments=0.1.2=pyh9f0ad1d_0
- jupyterlab_server=2.8.2=pyhd8ed1ab_0
- jupyterlab_widgets=1.0.2=pyhd8ed1ab_0
- k8=0.2.5=h9a82719_1
- kallisto=0.46.2=h60f4f9f_2
- keras-preprocessing=1.1.2=pyhd8ed1ab_0
- kernel-headers_linux-64=3.10.0=h4a8ded7_13
- kiwisolver=1.3.2=py39h1a9c180_1
- kmc=3.2.1=h9ee0642_0
- kmer-jellyfish=2.3.0=h7d875b9_2
- kraken2=2.1.2=pl5262h7d875b9_0
- krb5=1.19.2=hcc1bbae_3
- lcms2=2.12=hddcbb42_0
- ld_impl_linux-64=2.36.1=hea4e1c9_2
- ldc=1.20.0=h9a1ace1_1
- lerc=3.0=h9c3ff4c_0
- libarchive=3.5.2=hccf745f_1
- libassuan=2.5.5=h9c3ff4c_0
- libbigwig=0.4.6=h1b8c3c0_1
- libblas=3.9.0=16_linux64_openblas
- libbrotlicommon=1.0.9=h7f98852_6
- libbrotlidec=1.0.9=h7f98852_6
- libbrotlienc=1.0.9=h7f98852_6
- libcblas=3.9.0=16_linux64_openblas
- libcbor=0.8.0=h9c3ff4c_0
- libcurl=7.87.0=h91b91d3_0
- libdeflate=1.7=h7f98852_5
- libdivsufsort=2.0.2=h779adbc_4
- libedit=3.1.20210714=h7f8727e_0
- libev=4.33=h516909a_1
- libevent=2.1.10=h9b69904_4
- libfaiss=1.7.1=cuda112h5bea7ad_1_cuda
- libfaiss-avx2=1.7.1=cuda112h1234567_1_cuda
- libffi=3.4.2=h7f98852_5
- libfido2=1.9.0=h812cca2_1
- libgcc=7.2.0=h69d50b8_2
- libgcc-devel_linux-64=9.4.0=hd854feb_11
- libgcc-ng=12.2.0=h65d4601_19
- libgcrypt=1.10.1=h166bdaf_0
- libgd=2.3.2=h78a0170_0
- libgfortran=3.0.0=1
- libgfortran-ng=11.2.0=h69a702a_11
- libgfortran5=11.2.0=h5c6108e_11

- libglib=2.74.1=h7a41b64_0
- libgomp=12.2.0=h65d4601_19
- libgpg-error=1.45=hc0c96e0_0
- libgtextutils=0.7=h1b792b2_7
- libiconv=1.17=h166bdaf_0
- libidn2=2.3.2=h7f98852_0
- libjemalloc=5.2.1=h9c3ff4c_6
- libksba=1.3.5=hcb278e6_1001
- liblapack=3.9.0=16_linux64_openblas
- liblief=0.12.3=h27087fc_0
- libllvm10=10.0.1=he513fc3_3
- libllvm11=11.1.0=hf817b99_2
- libmamba=1.1.0=h2c5f835_2
- libmambapy=1.1.0=py39he50db72_2
- libnghttp2=1.46.0=hce63b2e_0
- libogg=1.3.5=h27cfd23_1
- libopenblas=0.3.21=pthreads_h78a6416_3
- libopus=1.3.1=h7f98852_1
- libpng=1.6.37=h21135ba_2
- libpq=13.5=hd57d9b9_0
- libprotobuf=3.16.0=h780b84a_0
- libsanitizer=9.4.0=h79bfe98_11
- libsodium=1.0.18=h36c2ea0_1
- libsolv=0.7.23=h3eb15da_0
- libsqlite=3.40.0=h753d276_0
- libssh2=1.10.0=ha56f1ee_2
- libstdcxx-devel_linux-64=9.4.0=hd854feb_11
- libstdcxx-ng=12.2.0=h46fd767_18
- libtiff=4.3.0=hf544144_0
- libudev1=249=h7f98852_1
- libunistring=0.9.10=h7f98852_0
- liburcu=0.13.2=h166bdaf_0
- libuuid=2.32.1=h7f98852_1000
- libuv=1.41.1=h7f98852_0
- libvorbis=1.3.7=h9c3ff4c_0
- libwebp=1.2.0=h3452ae3_0
- libwebp-base=1.2.0=h7f98852_2
- libxcb=1.14=h7b6447c_0
- libxkbcommon=1.0.3=he3ba5ed_0
- libxml2=2.9.12=h72842e0_0
- libxslt=1.1.33=h15afd5d_2
- libzlib=1.2.13=h166bdaf_4
- lightgbm=3.3.3=py39h5a03fae_1
- llvm-openmp=8.0.1=hc9558a2_0
- lml=0.1.0=pyh9f0ad1d_0
- locket=0.2.1=py39h06a4308_1
- lockfile=0.12.2=py_1
- lttng-ust=2.13.4=hfdfcbd3_0
- lz4-c=1.9.3=h9c3ff4c_1
- lzo=2.10=h516909a_1000
- mafft=7.520=hec16e2b_0
- magicblast=1.6.0=hf1761c0_1
- make=4.3=hd18ef5c_1
- mamba=1.1.0=py39hfa8f2c8_2
- mappy=2.23=py39hd1f1204_0
- markdown=3.3.6=pyhd8ed1ab_0
- markupsafe=2.0.1=py39h3811e60_1
- mash=2.3=he348c14_1
- mashmap=2.0=hd564ca7_6
- masurca=3.4.2=pl5262h86ccdc5_1
- matplotlib=3.5.0=py39hf3d152e_0
- matplotlib-base=3.5.0=py39h2fa2bec_0

```
- matplotlib-inline=0.1.3=pyhd8ed1ab_0
- matplotlib-venn=0.11.9=pyhd8ed1ab_0
- mawk=1.3.4=h779adbc_4
- megan=6.21.7=h9ee0642_0
- metaphlan=4.0.6=pyhca03a8a_0
- miniasm=0.3_r179=h5bf99c6_2
- minigraph=0.20=h7132678_0
- minimap2=2.24=h5bf99c6_0
- mira=4.9.6=1
- mistune=0.8.4=py39h3811e60_1005
- mmseqs2=13.45111=h95f258a_1
- more-itertools=8.12.0=pyhd8ed1ab_0
- moreutils=0.65=h7f98852_1
- mpc=1.2.1=h9f54685_0
- mpfr=4.1.0=h9202a9a_1
- mpi=1.0=openmpi
- mpmath=1.2.1=pyhd8ed1ab_0
- msamtools=1.1.3=he4a0461_0
- msgpack-python=1.0.3=py39h1a9c180_0
- multidict=5.2.0=py39h3811e60_1
- multiprocess=0.70.12.2=py39h3811e60_1
- multitasking=0.0.9=pyhd8ed1ab_0
- munkres=1.1.4=pyh9f0ad1d_0
- muscle=5.1=h9f5acd7_1
- mysql-common=8.0.29=haf5c9bc_1
- mysql-connector-c=6.1.11=h6eb9d5d_1007
- mysql-libs=8.0.29=h28c427c_1
- natsort=8.0.0=pyhd8ed1ab_0
- nbclassic=0.3.4=pyhd8ed1ab_0
- nbclient=0.5.9=pyhd8ed1ab_0
- nbconvert=6.3.0=py39hf3d152e_1
- nbformat=5.1.3=pyhd8ed1ab_0
- ncbi-datasets-cli=15.24.0=ha770c72_0
- ncbi-ngs-sdk=2.9.0=0
- ncdu=1.16=h0f457ee_0
- ncurses=6.4=h6a678d5_0
- nest-asyncio=1.5.1=pyhd8ed1ab_0
- networkx=2.6.3=pyhd8ed1ab_1
- ngmerge=0.3=ha92aebf_1
- nomkl=3.0=0
- notebook=6.4.6=pyha770c72_0
- npth=1.6=hf484d3e_1000
- nspr=4.32=h9c3ff4c_1
- nss=3.72=hb5efdd6_0
- ntbtls=0.1.2=hdbcaa40_1000
- numcodecs=0.9.1=py39he80948d_2
- numexpr=2.8.0=py39hbd72853_100
- numpy=1.24.3=py39h6183b62_0
- numpydoc=1.1.0=py_1
- oauthlib=3.1.1=pyhd8ed1ab_0
- odgi=0.6.3=py39h98c8e45_0
- olefile=0.46=pyh9f0ad1d_1
- oniguruma=6.9.8=h166bdaf_0
- openblas=0.3.21=pthreads_h320a7e8_3
- openjdk=11.0.1=h516909a_1016
- openjpeg=2.4.0=hb52868f_1
- openmp=8.0.1=0
- openmpi=4.1.2=hbfc84c5_0
- openpyxl=3.0.9=pyhd8ed1ab_0
- openssh=8.8p1=h1fa914a_1
- openssl=1.1.1w=hd590300_0
- opt_einsum=3.3.0=pyhd8ed1ab_1
```

- orfipy=0.0.4=py39h7cff6ad_0
- ossuuid=1.6.2=hf484d3e_1000
- packaging=21.3=pyhd8ed1ab_0
- pairix=0.3.7=py39hd1f1204_3
- paladin=1.4.6=h1b8c3c0_2
- pandas=1.3.4=py39hde0f152_1
- pandoc=2.16.2=h7f98852_0
- pandocfilters=1.5.0=pyhd8ed1ab_0
- pango=1.48.10=h54213e6_2
- parallel=20230322=ha770c72_0
- parallel-virfinder=0.3.1=py310hdfd78af_0
- param=1.12.0=pyh6c4a22f_0
- paramiko=3.3.1=pyhd8ed1ab_0
- parasail-python=1.2.4=py39h98c8e45_1
- parso=0.8.2=pyhd8ed1ab_0
- partd=1.2.0=pyhd8ed1ab_0
- patch=2.7.6=h7f98852_1002
- patchelf=0.17.2=h58526e2_0
- pathracer=3.16.0.dev=h95f258a_0
- patsy=0.5.2=pyhd8ed1ab_0
- pbzip2=1.1.13=0
- pcre=8.45=h9c3ff4c_0
- pcre2=10.37=h032f7d1_0
- perl=5.26.2=h36c2ea0_1008
- perl-apache-test=1.40=pl526_1
- perl-app-cpanminus=1.7044=pl526_1
- perl-archive-tar=2.32=pl526_0
- perl-base=2.23=pl526_1
- perl-business-isbn=3.004=pl526_0
- perl-business-isbn-data=20140910.003=pl526_0
- perl-carp=1.38=pl526_3
- perl-class-load=0.25=pl526_0
- perl-class-load-xs=0.10=pl526h6bb024c_2
- perl-class-method-modifiers=2.12=pl526_0
- perl-common-sense=3.74=pl526_2
- perl-compress-raw-bzip2=2.087=pl526he1b5a44_0
- perl-compress-raw-zlib=2.087=pl526hc9558a2_0
- perl-constant=1.33=pl526_1
- perl-data-dumper=2.173=pl526_0
- perl-data-optlist=0.110=pl526_2
- perl-devel-globaldestruction=0.14=pl526_0
- perl-devel-overloadinfo=0.005=pl526_0
- perl-devel-stacktrace=2.04=pl526_0
- perl-digest-hmac=1.03=pl526_3
- perl-digest-md5=2.55=pl526_0
- perl-dist-checkconflicts=0.11=pl526_2
- perl-encode=2.88=pl526_1
- perl-encode-locale=1.05=pl526_6
- perl-eval-closure=0.14=pl526h6bb024c_4
- perl-exporter=5.72=pl526_1
- perl-exporter-tiny=1.002001=pl526_0
- perl-extutils-makemaker=7.36=pl526_1
- perl-file-listing=6.04=pl526_1
- perl-file-path=2.16=pl526_0
- perl-file-temp=0.2304=pl526_2
- perl-filesys-df=0.92=pl526h14c3975_3
- perl-gd=2.68=pl526he941832_0
- perl-gdgraph=1.54=pl526_0
- perl-gdgraph-histogram=1.1=pl526_3
- perl-gdtextutil=0.86=pl526h14c3975_5
- perl-getopt-long=2.50=pl526_1
- perl-html-parser=3.72=pl526h6bb024c_5

- perl-html-tagset=3.20=pl526_3
- perl-html-tree=5.07=pl526_1
- perl-http-cookies=6.04=pl526_0
- perl-http-daemon=6.01=pl526_1
- perl-http-date=6.02=pl526_3
- perl-http-message=6.18=pl526_0
- perl-http-negotiate=6.01=pl526_3
- perl-io-compress=2.087=pl526he1b5a44_0
- perl-io-handle=1.36=pl526_1
- perl-io-html=1.001=pl526_2
- perl-io-socket-ssl=2.066=pl526_0
- perl-io-zlib=1.10=pl526_2
- perl-json=4.02=pl526_0
- perl-json-xs=2.34=pl526h6bb024c_3
- perl-libwww-perl=6.39=pl526_0
- perl-list-moreutils=0.428=pl526_1
- perl-list-moreutils-xs=0.428=pl526_0
- perl-list-util=1.38=pl526_1
- perl-lwp-mediatypes=6.04=pl526_0
- perl-lwp-protocol-https=6.07=pl526_4
- perl-mime-base64=3.15=pl526_1
- perl-module-implementation=0.09=pl526_2
- perl-module-runtime=0.016=pl526_1
- perl-module-runtime-conflicts=0.003=pl526_0
- perl-moo=2.003004=pl526_0
- perl-moose=2.2011=pl526hf484d3e_1
- perl-mozilla-ca=20180117=pl526_1
- perl-mro-compat=0.13=pl526_0
- perl-net-http=6.19=pl526_0
- perl-net-ssleay=1.88=pl526h90d6eec_0
- perl-ntlm=1.09=pl526_4
- perl-package-deprecationmanager=0.17=pl526_0
- perl-package-stash=0.38=pl526hf484d3e_1
- perl-package-stash-xs=0.28=pl526hf484d3e_1
- perl-parallel-forkmanager=2.02=pl526_0
- perl-params-util=1.07=pl526h6bb024c_4
- perl-parent=0.236=pl526_1
- perl-pathtools=3.75=pl526h14c3975_1
- perl-perlio-gzip=0.20=pl526h84994c4_1
- perl-role-tiny=2.000008=pl526_0
- perl-scalar-list-utils=1.52=pl526h516909a_0
- perl-socket=2.027=pl526_1
- perl-storable=3.15=pl526h14c3975_0
- perl-sub-exporter=0.987=pl526_2
- perl-sub-exporter-progressive=0.001013=pl526_0
- perl-sub-identify=0.14=pl526h14c3975_0
- perl-sub-install=0.928=pl526_2
- perl-sub-name=0.21=pl526_1
- perl-sub-quote=2.006003=pl526_1
- perl-test-requiresinternet=0.05=pl526_0
- perl-time-local=1.28=pl526_1
- perl-try-tiny=0.30=pl526_1
- perl-types-serialiser=1.0=pl526_2
- perl-uri=1.76=pl526_0
- perl-vcftools-vcf=0.1.16=pl526_2
- perl-www-robotrules=6.02=pl526_3
- perl-xml-libxml=2.0132=pl526h7ec2d77_1
- perl-xml-namespacesupport=1.12=pl526_0
- perl-xml-parser=2.44_01=pl15262hc3e0081_1002
- perl-xml-sax=1.02=pl526_0
- perl-xml-sax-base=1.09=pl526_0
- perl-xml-sax-expat=0.51=pl526_3

- perl-xml-simple=2.25=pl526_1
- perl-xsloader=0.24=pl526_0
- pexpect=4.8.0=pyh9f0ad1d_2
- phylophlan=3.0.3=pyhdfd78af_0
- pickleshare=0.7.5=py39hde42818_1002
- pigz=2.6=h27826a3_0
- pillow=8.4.0=py39h5aabda8_0
- pip=21.3.1=pyhd8ed1ab_0
- pixman=0.40.0=h36c2ea0_0
- pkginfo=1.9.6=pyhd8ed1ab_0
- plass=4.687d7=h95f258a_2
- plink=1.90b6.21=h779adbc_1
- plotly=5.5.0=pyhd8ed1ab_0
- pluggy=1.0.0=py39hf3d152e_2
- pomegranate=0.14.4=py39h9a67853_0
- popt=1.16=1
- poyo=0.5.0=py_0
- prince-factor-analysis=0.7.1=pyhd8ed1ab_1
- prodigal=2.6.3=hec16e2b_4
- prometheus_client=0.12.0=pyhd8ed1ab_0
- prompt-toolkit=3.0.22=pyha770c72_0
- prompt_toolkit=3.0.22=hd8ed1ab_0
- protobuf=3.16.0=py39he80948d_0
- psutil=5.8.0=py39h3811e60_2
- pthread-stubs=0.4=h36c2ea0_1001
- ptyprocess=0.7.0=pyhd3deb0d_0
- pulp=2.6.0=py39hf3d152e_0
- py=1.11.0=pyh6c4a22f_0
- py-lief=0.12.3=py39h5a03fae_0
- py2bit=0.3.0=py39h38f01e4_5
- pyahocorasick=1.4.0=py39h07f9747_2
- pyasn1=0.4.8=py_0
- pyasn1-modules=0.2.8=py_0
- pybbi=0.3.0=py39h544ab2f_1
- pybigwig=0.3.18=py39h015b436_1
- pybind11=2.11.1=py39h7633fee_0
- pybind11-abi=4=hd8ed1ab_3
- pybind11-global=2.11.1=py39h7633fee_0
- pycosat=0.6.3=py39h3811e60_1009
- pycparser=2.21=pyhd8ed1ab_0
- pyct=0.4.6=py_0
- pyct-core=0.4.6=py_0
- pyexcel=0.6.7=pyhd8ed1ab_0
- pyexcel-ezodf=0.3.4=py_0
- pyexcel-io=0.6.4=pyhd8ed1ab_0
- pyexcel-ods3=0.5.3=py_1
- pyexcel-xls=0.6.1=pyh9f0ad1d_0
- pyfaidx=0.6.3.1=pyh5e36f6f_0
- pyfastx=0.8.4=py39hd1f1204_0
- pygments=2.10.0=pyhd8ed1ab_0
- pyjwt=2.3.0=pyhd8ed1ab_0
- pynacl=1.5.0=py39hd1e30aa_3
- pyopenssl=21.0.0=pyhd8ed1ab_0
- pyparsing=3.0.6=pyhd8ed1ab_0
- pyqt=5.12.3=py39hf3d152e_8
- pyqt-impl=5.12.3=py39hde8b62d_8
- pyqt5-sip=4.19.18=py39he80948d_8
- pyqtchart=5.12=py39h0fcd23e_8
- pyqtwebengine=5.12.1=py39h0fcd23e_8
- pyrate-limiter=2.9.0=pyhd8ed1ab_0
- pyrsistent=0.18.0=py39h3811e60_0
- pysam=0.17.0=py39h051187c_0

```
- pysocks=1.7.1=py39hf3d152e_4
- pytest=6.2.5=py39hf3d152e_1
- python=3.9.16=h7a1cb2a_2
- python-annoy=1.17.0=py39he80948d_3
- python-dateutil=2.8.2=pyhd8ed1ab_0
- python-libarchive-c=4.0=py39hf3d152e_1
- python-lzo=1.12=py39h265373d_1004
- python_abi=3.9=2_cp39
- pytz=2021.3=pyhd8ed1ab_0
- pyu2f=0.1.5=pyhd8ed1ab_0
- pyyaml=6.0=py39h3811e60_3
- pyzmq=22.3.0=py39h37b5a0c_1
- qt=5.12.9=hda022c4_4
- qtconsole=5.2.1=pyhd8ed1ab_0
- qtpy=1.11.3=pyhd8ed1ab_0
- r-amap=0.8_19=r41hb13c81a_0
- r-ape=5.6_2=r41h9f5de39_1
- r-argparse=2.1.6=r41hc72bb7e_1
- r-ashr=2.2_54=r41h7525677_1
- r-askpass=1.1=r41h06615bd_3
- r-assertthat=0.2.1=r41hc72bb7e_3
- r-backports=1.4.1=r41h06615bd_1
- r-base=4.1.1=hb67fd72_0
- r-base64enc=0.1_3=r41h06615bd_1005
- r-bbmle=1.0.25=r41hc72bb7e_1
- r-bdsmatrix=1.3_6=r41h06615bd_1
- r-bh=1.78.0_0=r41hc72bb7e_1
- r-biasedurn=2.0.8=r41h7525677_0
- r-bit=4.0.4=r41h06615bd_1
- r-bit64=4.0.5=r41h06615bd_1
- r-bitops=1.0_7=r41h06615bd_1
- r-blob=1.2.3=r41hc72bb7e_1
- r-brio=1.1.3=r41h06615bd_1
- r-broom=1.0.1=r41hc72bb7e_1
- r-bslib=0.4.1=r41hc72bb7e_0
- r-cachem=1.0.6=r41h06615bd_1
- r-callr=3.7.3=r41hc72bb7e_0
- r-catools=1.18.2=r41h7525677_1
- r-cellranger=1.1.0=r41hc72bb7e_1005
- r-cli=3.4.1=r41h7525677_1
- r-clipr=0.8.0=r41hc72bb7e_1
- r-cluster=2.1.4=r41h8da6f51_0
- r-coda=0.19_4=r41hc72bb7e_1
- r-codetools=0.2_19=r41hc72bb7e_0
- r-colorspace=2.0_3=r41h06615bd_1
- r-commonmark=1.8.1=r41h06615bd_0
- r-cpp11=0.4.3=r41hc72bb7e_0
- r-crayon=1.5.2=r41hc72bb7e_1
- r-curl=4.3.3=r41h06615bd_1
- r-data.table=1.14.4=r41h06615bd_0
- r-dbi=1.1.3=r41hc72bb7e_1
- r-dbplyr=2.2.1=r41hc72bb7e_1
- r-desc=1.4.2=r41hc72bb7e_1
- r-diffobj=0.3.5=r41h06615bd_1
- r-digest=0.6.30=r41h7525677_0
- r-dplyr=1.0.10=r41h7525677_1
- r-dtplyr=1.2.2=r41hc72bb7e_1
- r-ellipsis=0.3.2=r41h06615bd_1
- r-emdbook=1.3.12=r41hc72bb7e_2
- r-etrunct=0.1=r41hc72bb7e_1004
- r-evaluate=0.18=r41hc72bb7e_0
- r-fansi=1.0.3=r41h06615bd_1
```

```
- r-farver=2.1.1=r41h7525677_1
- r-fastcluster=1.2.3=r41h27087fc_1
- r-fastmap=1.1.0=r41h7525677_1
- r-fastmatch=1.1_3=r41h06615bd_1
- r-filelock=1.0.2=r41h06615bd_1003
- r-findpython=1.0.7=r41hc72bb7e_1
- r-forcats=0.5.2=r41hc72bb7e_1
- r-foreach=1.5.2=r41hc72bb7e_1
- r-formatr=1.12=r41hc72bb7e_1
- r-fs=1.5.2=r41h7525677_2
- r-futile.logger=1.4.3=r41hc72bb7e_1004
- r-futile.options=1.0.1=r41hc72bb7e_1003
- r-gargle=1.2.1=r41hc72bb7e_1
- r-generics=0.1.3=r41hc72bb7e_1
- r-ggplot2=3.4.0=r41hc72bb7e_0
- r-glmnet=4.1_2=r41h8da6f51_1
- r-glue=1.6.2=r41h06615bd_1
- r-googledrive=2.0.0=r41hc72bb7e_1
- r-googlesheets4=1.0.1=r41h785f33e_1
- r-gplots=3.1.3=r41hc72bb7e_1
- r-gtable=0.3.1=r41hc72bb7e_1
- r-gtools=3.9.3=r41h06615bd_1
- r-haven=2.5.1=r41h7525677_0
- r-highr=0.9=r41hc72bb7e_1
- r-hms=1.1.2=r41hc72bb7e_1
- r-htmltools=0.5.3=r41h7525677_1
- r-httr=1.4.4=r41hc72bb7e_1
- r-hwriter=1.3.2.1=r41hc72bb7e_1
- r-ids=1.0.1=r41hc72bb7e_2
- r-igraph=1.3.0=r41hf10d5bd_0
- r-invgamma=1.1=r41hc72bb7e_2
- r-irlba=2.3.5.1=r41h5f7b363_0
- r-isoband=0.2.6=r41h7525677_1
- r-iterators=1.0.14=r41hc72bb7e_1
- r-jquerylib=0.1.4=r41hc72bb7e_1
- r-jsonlite=1.8.3=r41h06615bd_0
- r-kernsmooth=2.23_20=r41hd009a43_1
- r-knitr=1.40=r41hc72bb7e_1
- r-labeling=0.4.2=r41hc72bb7e_2
- r-lambda.r=1.2.4=r41hc72bb7e_2
- r-lattice=0.20_45=r41h06615bd_1
- r-lifecycle=1.0.3=r41hc72bb7e_1
- r-locfit=1.5_9.6=r41h06615bd_1
- r-lubridate=1.8.0=r41h7525677_1
- r-magrittr=2.0.3=r41h06615bd_1
- r-markdown=1.3=r41hc72bb7e_0
- r-mass=7.3_58.1=r41h06615bd_1
- r-matrix=1.5_3=r41h5f7b363_0
- r-matrixstats=0.62.0=r41h06615bd_1
- r-memoise=2.0.1=r41hc72bb7e_1
- r-mgcv=1.8_41=r41h5f7b363_0
- r-mime=0.12=r41h06615bd_1
- r-mixsqp=0.3_43=r41h9f5de39_2
- r-modelr=0.1.10=r41hc72bb7e_0
- r-munsell=0.5.0=r41hc72bb7e_1005
- r-mvtnorm=1.1_3=r41h8da6f51_1
- r-nlme=3.1_160=r41h8da6f51_0
- r-numderiv=2016.8_1.1=r41hc72bb7e_4
- r-openssl=2.0.4=r41hfaab4ff_0
- r-phangorn=2.9.0=r41h37cf8d7_1
- r-pillar=1.8.1=r41hc72bb7e_1
- r-pkgconfig=2.0.3=r41hc72bb7e_2
```

```
- r-pkgload=1.3.1=r41hc72bb7e_0
- r-plogr=0.2.0=r41hc72bb7e_1004
- r-plyr=1.8.8=r41h7525677_0
- r-png=0.1_7=r41h06615bd_1005
- r-praise=1.0.0=r41hc72bb7e_1006
- r-prettyunits=1.1.1=r41hc72bb7e_2
- r-processx=3.8.0=r41h06615bd_0
- r-progress=1.2.2=r41hc72bb7e_3
- r-ps=1.7.2=r41h06615bd_0
- r-purrr=0.3.5=r41h06615bd_1
- r-quadprog=1.5_8=r41hd009a43_4
- r-r6=2.5.1=r41hc72bb7e_1
- r-rappdirs=0.3.3=r41h06615bd_1
- r-rcolorbrewer=1.1_3=r41h785f33e_1
- r-rcpp=1.0.9=r41h7525677_2
- r-rcpparmadillo=0.11.4.2.1=r41h9f5de39_0
- r-rcppeigen=0.3.3.9.3=r41h9f5de39_0
- r-rcppnumerical=0.4_0=r41h7525677_2
- r-rcurl=1.98_1.5=r41hcfec24a_0
- r-readr=2.1.3=r41h7525677_1
- r-readxl=1.4.1=r41hf23e330_0
- r-rematch=1.0.1=r41hc72bb7e_1005
- r-rematch2=2.1.2=r41hc72bb7e_2
- r-reprex=2.0.2=r41hc72bb7e_1
- r-reshape2=1.4.4=r41h7525677_2
- r-restfulr=0.0.15=r41h73dbb54_0
- r-rjson=0.2.21=r41h7525677_2
- r-rlang=1.0.6=r41h7525677_1
- r-rmarkdown=2.18=r41hc72bb7e_0
- r-rprojroot=2.0.3=r41hc72bb7e_1
- r-rsqlite=2.2.18=r41h7525677_0
- r-rstudioapi=0.14=r41hc72bb7e_1
- r-rvest=1.0.3=r41hc72bb7e_1
- r-sass=0.4.2=r41h7525677_1
- r-scales=1.2.1=r41hc72bb7e_1
- r-selectr=0.4_2=r41hc72bb7e_2
- r-shape=1.4.6=r41ha770c72_1
- r-sm=2.2_5.7.1=r41h8da6f51_1
- r-snow=0.4_4=r41hc72bb7e_1
- r-squarem=2021.1=r41hc72bb7e_1
- r-statmod=1.4.37=r41hc3ea6d6_1
- r-stringi=1.7.6=r41hcabe038_0
- r-stringr=1.4.1=r41hc72bb7e_1
- r-survival=3.4_0=r41h06615bd_1
- r-sys=3.4.1=r41h06615bd_0
- r-testthat=3.1.5=r41h7525677_1
- r-tibble=3.1.8=r41h06615bd_1
- r-tidyr=1.2.1=r41h7525677_1
- r-tidyselect=1.2.0=r41hc72bb7e_0
- r-tidyverse=1.3.2=r41hc72bb7e_1
- r-tinytex=0.42=r41hc72bb7e_1
- r-truncnorm=1.0_8=r41h06615bd_1003
- r-tzdb=0.3.0=r41h7525677_1
- r-utf8=1.2.2=r41h06615bd_1
- r-uuid=1.1_0=r41h06615bd_1
- r-vctrs=0.5.0=r41h7525677_0
- r-vioplot=0.3.7=r41hc72bb7e_1
- r-virfinder=1.1=r41h87f3376_4
- r-viridislite=0.4.1=r41hc72bb7e_1
- r-vroom=1.6.0=r41h7525677_1
- r-waldo=0.4.0=r41hc72bb7e_1
- r-withr=2.5.0=r41hc72bb7e_1
```

```
- r-xfun=0.34=r41h7525677_0
- r-xml=3.99_0.9=r41h06615bd_0
- r-xml2=1.3.3=r41h03ef668_0
- r-xtable=1.8_4=r41hc72bb7e_4
- r-yaml=2.3.6=r41h06615bd_0
- r-zoo=1.8_11=r41h06615bd_1
- rapidnj=2.3.2=h7d875b9_1
- ratelimiter=1.2.0=py_1002
- raxml=8.2.12=hec16e2b_4
- re2=2021.09.01=h9c3ff4c_0
- readline=8.2=h5eee18b_0
- reproc=14.2.3=h7f98852_0
- reproc-cpp=14.2.3=h9c3ff4c_0
- requests=2.26.0=pyhd8ed1ab_1
- requests-cache=0.9.8=pyhd8ed1ab_0
- requests-oauthlib=1.3.0=pyh9f0ad1d_0
- requests-ratelimiter=0.4.0=pyhd8ed1ab_0
- requests-unixsocket=0.2.0=py_0
- reseq=1.1=py39hdced79b_1
- rhash=1.4.1=h7f98852_0
- rich=12.4.1=pyhd8ed1ab_0
- ripgrep=13.0.0=h2f28480_2
- rnacode=0.3=h779adbc_2
- rnaz=2.1.1=pl5262h1b792b2_2
- rsa=4.8=pyhd8ed1ab_0
- rsync=3.2.3=hfa40b15_4
- ruamel.yaml=0.16.12=py39h3811e60_2
- ruamel.yaml.clib=0.2.6=py39h3811e60_0
- ruamel_yaml=0.15.100=py39h27cfd23_0
- ruptures=1.1.5=py39hce5d2b2_1
- salmon=1.5.2=h84f40af_0
- sambamba=0.6.8=h682856c_0
- samtools=1.15=h3843a85_0
- scikit-allel=1.3.5=py39hde0f152_1
- scikit-bio=0.5.6=py39h16ac069_4
- scikit-learn=1.0.1=py39h4dfa638_2
- scipy=1.9.3=py39h32ae08f_0
- seaborn=0.11.2=hd8ed1ab_0
- seaborn-base=0.11.2=pyhd8ed1ab_0
- sed=4.8=he412f7d_0
- send2trash=1.8.0=pyhd8ed1ab_0
- seqfu=1.17.1=hbd632db_0
- seqkit=2.1.0=h9ee0642_0
- seqtk=1.3=h5bf99c6_3
- setuptools=59.4.0=py39hf3d152e_0
- sga=0.10.15=ha89c123_6
- shap=0.40.0=py39hde0f152_0
- shorah=1.99.2=py39h4bc2be3_3
- shustring=2.6=h779adbc_4
- sickle-trim=1.33=h5bf99c6_6
- simplejson=3.17.6=py39h3811e60_0
- six=1.15.0=pyh9f0ad1d_0
- sklearn-contrib-py-earth=0.1.0=py39h16ac069_3
- slicer=0.0.7=pyhd8ed1ab_0
- smmap=3.0.5=pyh44b312d_0
- snakemake-minimal=5.26.0=py_0
- snappy=1.1.8=he1b5a44_3
- sniffio=1.2.0=py39hf3d152e_2
- snowballstemmer=2.2.0=pyhd8ed1ab_0
- snpgenie=1.0=hdfd78af_1
- sortedcontainers=2.4.0=pyhd8ed1ab_0
- soupsieve=2.3.2.post1=pyhd8ed1ab_0
```

```
- spades=3.15.5=h95f258a_1
- sparsehash=2.0.4=h9c3ff4c_0
- sphinx=4.3.1=pyh6c4a22f_0
- sphinxcontrib-applehelp=1.0.2=py_0
- sphinxcontrib-devhelp=1.0.2=py_0
- sphinxcontrib-htmlhelp=2.0.0=pyhd8ed1ab_0
- sphinxcontrib-jsmath=1.0.1=py_0
- sphinxcontrib-qthelp=1.0.3=py_0
- sphinxcontrib-serializinghtml=1.1.5=pyhd8ed1ab_1
- sqlite=3.40.1=h5082296_0
- squizz=0.99d=h779adbc_3
- sra-tools=2.11.0=pl5262h314213e_1
- statsmodels=0.13.1=py39hce5d2b2_0
- svgutils=0.3.4=pyhd8ed1ab_0
- swipe=2.1.1=h2152503_0
- sympy=1.9=py39hf3d152e_1
- sysroot_linux-64=2.17=h4a8ded7_13
- tar=1.34=ha1f6473_0
- tbb=2020.3=hfd86e86_0
- tblib=1.7.0=pyhd8ed1ab_0
- tedna=1.2.2=hfc679d8_2
- tenacity=8.0.1=pyhd8ed1ab_0
- tensorboard-data-server=0.6.0=py39h95dcef6_1
- tensorboard-plugin-wit=1.8.0=pyh44b312d_0
- tensorflow=2.6.0=cpu_py39hcb7c6aa_2
- tensorflow-base=2.6.0=cpu_py39h7e79a0b_2
- tensorflow-probability=0.15.0=pyhd8ed1ab_0
- termcolor=1.1.0=py_2
- terminado=0.12.1=py39hf3d152e_1
- testpath=0.5.0=pyhd8ed1ab_0
- texttable=1.6.4=pyhd8ed1ab_0
- threadpoolctl=3.0.0=pyh8a188c0_0
- tk=8.6.12=h27826a3_0
- tktable=2.10=hb7b940f_3
- toml=0.10.2=pyhd8ed1ab_0
- toolz=0.11.2=pyhd8ed1ab_0
- toposort=1.7=pyhd8ed1ab_0
- tornado=6.1=py39h3811e60_2
- tqdm=4.65.0=pyhd8ed1ab_0
- traitlets=5.1.1=pyhd8ed1ab_0
- trimal=1.4.1=h9f5acd7_6
- trimmomatic=0.39=hdfd78af_2
- trinity=2.13.2=h00214ad_1
- twopaco=0.9.4=he711bca_1
- typing-extensions=4.7.1=hd8ed1ab_0
- typing_extensions=4.7.1=pyha770c72_0
- tzdata=2021e=he74cb21_0
- ucsc-bedgraphtobigwig=377=h0b8a92a_2
- ucsc-bigwigtobedgraph=377=h0b8a92a_5
- ucsc-fatotwobit=377=h0b8a92a_4
- ucsc-twobittofa=377=h0b8a92a_3
- ucsc-wigtobigwig=377=h0b8a92a_2
- ujson=5.7.0=py39h227be39_0
- upsetplot=0.8.0=pyhd8ed1ab_0
- url-normalize=1.4.3=pyhd8ed1ab_0
- urllib3=1.26.7=pyhd8ed1ab_0
- vcftools=0.1.16=h9a82719_5
- vg=1.37.0=h9ee0642_0
- vibrant=1.0.1=py37h5ca1d4c_1
- virsorter=2.2.3=pyhdfd78af_1
- voila=0.2.11=pyhd8ed1ab_0
- wcwidth=0.2.5=pyh9f0ad1d_2
```

```
- webencodings=0.5.1=py_1
- websocket-client=1.2.1=py39hf3d152e_0
- werkzeug=2.0.2=pyhd3eb1b0_0
- wget=1.20.3=ha56f1ee_1
- wheel=0.37.0=pyhd8ed1ab_1
- whichcraft=0.6.1=py_0
- widgetsnbextension=3.5.2=py39hf3d152e_1
- wiggletools=1.2.1=0
- wquantiles=0.6=pyhd8ed1ab_0
- wrapt=1.12.1=py39h3811e60_3
- xlrd=2.0.1=pyhd8ed1ab_3
- xlwt=1.3.0=py_1
- xorg-fixesproto=5.0=h7f98852_1002
- xorg-inputproto=2.3.2=h7f98852_1002
- xorg-kbproto=1.0.7=h7f98852_1002
- xorg-libice=1.0.10=h7f98852_0
- xorg-libsm=1.2.3=hd9c2040_1000
- xorg-libx11=1.7.2=h7f98852_0
- xorg-libxau=1.0.9=h7f98852_0
- xorg-libxdmcp=1.1.3=h7f98852_0
- xorg-libxext=1.3.4=h7f98852_1
- xorg-libxfixes=5.0.3=h7f98852_1004
- xorg-libxi=1.7.10=h7f98852_0
- xorg-libxrender=0.9.10=h7f98852_1003
- xorg-libxt=1.2.1=h7f98852_2
- xorg-libxtst=1.2.3=h7f98852_1002
- xorg-recordproto=1.14.2=h7f98852_1002
- xorg-renderproto=0.11.1=h7f98852_1002
- xorg-xextproto=7.3.0=h7f98852_1002
- xorg-xproto=7.0.31=h7f98852_1007
- xxhash=0.8.0=h7f98852_3
- xz=5.2.10=h5eee18b_1
- yaml=0.2.5=h516909a_0
- yaml-cpp=0.7.0=h27087fc_1
- yarl=1.7.2=py39h3811e60_1
- yfinance=0.2.14=pyhd8ed1ab_0
- zarr=2.10.3=pyhd8ed1ab_0
- zeromq=4.3.4=h9c3ff4c_1
- zict=2.0.0=py_0
- zipp=3.6.0=pyhd8ed1ab_0
- zlib=1.2.13=h166bdaf_4
- zstd=1.5.2=h8a70e8d_1
- pip:
  - absl-py==1.3.0
  - automat==22.10.0
  - chess==1.9.4
  - constantly==15.1.0
  - entrezpy==2.1.3
  - filtersam==0.0.8
  - fit-nbinom==1.1
  - flatbuffers==22.10.26
  - grpcio==1.34.1
  - h11==0.14.0
  - httpcore==0.16.3
  - httpx==0.23.3
  - hyperlink==21.0.0
  - incremental==22.10.0
  - investiny==0.7.2
  - investpy==1.0.8
  - keras==2.4.3
  - libclang==14.0.6
  - llvmlite==0.40.0
```

```
  - lxml==4.9.2
  - mbtr==0.1.3
  - mca==1.0.3
  - memory-profiler==0.61.0
  - ncbi-taxonomist==1.2.1
  - numba==0.57.0
  - parallelbam==0.0.18
  - pydantic==1.10.11
  - python-chess==1.999
  - requests-futures==1.0.0
  - rfc3986==1.5.0
  - rotation-forest==1.0
  - scann==1.2.2
  - stockfish==3.28.0
  - tcod==15.0.0
  - tensorboard==2.10.1
  - tensorflow-cpu==2.10.0
  - tensorflow-estimator==2.10.0
  - tensorflow-io-gcs-filesystem==0.27.0
  - twisted==22.10.0
  - unidecode==1.3.6
  - yahooquery==2.3.1
  - zope-interface==6.0
```

Appendix B. Mottle Program Code

```
import argparse
import numpy as np

parser = argparse.ArgumentParser(
    prog = 'mottle.py',
    description = 'Mottle - Pairwise substitution distance at high divergences',
    epilog = 'Mottle Copyright 2023 Newcastle University. All Rights Reserved.\n' +
    'Authors: Alisa Prusokiene, Neil Boonham, Adrian Fox, and Thomas P. Howard.\n' +
    'The initial repository for this software is located at https://github.com/tphoward/Mottle_Repo.',
    formatter_class=argparse.ArgumentDefaultsHelpFormatter)
parser.add_argument('in1_p', nargs='?', type=argparse.FileType('r'), default=None, metavar='in1',
                    help='Input fasta file 1')
parser.add_argument('in2_p', nargs='?', type=argparse.FileType('r'), default='-', metavar='in2',
                    help='Input fasta file 2')
parser.add_argument('out_p', nargs='?', type=argparse.FileType('w'), default='-', metavar='out',
                    help='Output file for final value')
parser.add_argument('-i', '--in1', '--in', type=argparse.FileType('r'), default=None,
                    help='Input fasta file 1')
parser.add_argument('-I', '--in2', '-j', type=argparse.FileType('r'), default=None,
                    help='Input fasta file 2')
parser.add_argument('-o', '--out', type=argparse.FileType('w'), default=None,
                    help='Output file for final value')
parser.add_argument('--chunk_size', type=np.uint, default=18,
                    help='Size of each chunk in base pairs')
parser.add_argument('--nchunks', type=np.uint, default=20,
                    help='Total number of chunks used for search window')
parser.add_argument('--guide_chunks', type=np.uint, default=5,
                    help='Number of chunks directly aligned, without InDels taken into account')
parser.add_argument('--window_shape', type=str, default='boxcar', choices=['boxcar'],
                    help='Shape of search window')
parser.add_argument('--encodings', type=str, nargs='*', default=['MAFFT'],
                    choices=['MAFFT', 'SQUARE', 'AG', 'PUPY', 'TRANS', 'NUC.4.4', 'BLOSUM62', 'PROPS'],
                    help='Encodings used for sequence search')
parser.add_argument('--detr', type=str, default='constant', choices=['constant'],
                    help='GC content correction mode')
parser.add_argument('--norm_level', type=np.uint, default=1,
                    help='Exponent used for normalisation')
parser.add_argument('--index', type=str, default='Flat', choices=['Flat'],
                    help='Faiss search index type')
parser.add_argument('--reduct', type=str, default='mip', choices=['mip'],
                    help='Faiss search reduction type')
parser.add_argument('--nmatch', type=np.uint, nargs='*', default=[1],
                    help='Number of nearest neighbours returned per site')
parser.add_argument('--sample', type=str, default='all', choices=['all'],
                    help='Subsample fraction')
parser.add_argument('--filt_size', type=np.uint, default=9,
                    help='Size of window for filtering near-origin alignment gaps')
parser.add_argument('--max_pass', type=np.uint, default=3000,
                    help='Maximum windows kept after filtering')
parser.add_argument('--cut_thres', type=np.uint, default=2,
                    help='Maximum sub-window divergence before alignment is cut')
```

```
parser.add_argument('--samp_width', type=np.uint, default=100,
                    help='Sub-window sample size')
parser.add_argument('--min_samps', type=np.uint, default=150,
                    help='Minimum number of sub-windows samples so that alignment is not discarded')
parser.add_argument('--ntrees', type=np.uint, default=100,
                    help='LightGBM number of trees for true identity estimation')
parser.add_argument('--nleaves', type=np.uint, default=31,
                    help='LightGBM number of leaves')
parser.add_argument('--learn_rate', type=np.uint, default=0.1,
                    help='LightGBM learn rate')
parser.add_argument('--subsamp', type=np.uint, default=1,
                    help='LightGBM subsample proportion')
parser.add_argument('--binpow', type=np.uint, default=64,
                    help='Exponent for cluster discretisation')
parser.add_argument('--learn_mult', type=np.uint, default=0.001,
                    help='Tensorflow learning multiplier')
parser.add_argument('--reltol', type=np.float_, default=1e-20,
                    help='Tensorflow relative tolerace stopping codition')
parser.add_argument('--maxiter', type=np.uint, default=100,
                    help='Tensorflow maximum number of gradient descent iterations')
parser.add_argument('--binthres', type=np.uint, default=0.75,
                    help='Threshold for sequence inclusion into homology cluster')
parser.add_argument('--prior_size', type=np.uint, default=10,
                    help='Bias value for distance calculations where few alignments pass the filtering stage')
parser.add_argument('--ncpu', type=np.uint, default=4,
                    help='Number of cpu threads for intensive tasks')
parser.add_argument('-v', '--verbose', action='store_true',
                    help='Verbosity toggle')

args = parser.parse_args()
infile1 = args.in1 if args.in1 else args.in1_p
infile2 = args.in2 if args.in2 else args.in2_p
outfile = args.out if args.out else args.out_p
if infile1 is None:
    parser.print_help()
    exit()

chunk_size = args.chunk_size
nchunks = args.nchunks
guide_chunks = args.guide_chunks
window_size = nchunks * chunk_size
guide_size = guide_chunks * chunk_size
window_shape = args.window_shape
encodings = args.encodings
detr = args.detr
norm_level = args.norm_level
index = args.index
reduct = args.reduct
nmatch = args.nmatch
sample = args.sample
filt_size = args.filt_size
max_pass = args.max_pass
cut_thres = args.cut_thres
samp_width = args.samp_width
min_samps = args.min_samps
ntrees = args.ntrees
nleaves = args.nleaves
learn_rate = args.learn_rate
subsamp = args.subsamp
binpow = args.binpow
learn_mult = args.learn_mult
reltol = args.reltol
```

```
maxiter = args.maxiter
binthres = args.binthres
prior_size = args.prior_size
ncpu = args.ncpu
verbose = args.verbose

eps = np.finfo(np.float32).resolution

import sys, re

from collections import defaultdict
from io import StringIO
from Bio import SeqIO
from scipy import signal, fft, stats
from numpy import ma
from numpy.lib.stride_tricks import as_strided
import faiss
import parasail
from tqdm import tqdm
import lightgbm as lgb
import tensorflow as tf
import tensorflow_probability as tfp

def strtoseq(string):
    """Convert string to numpy array."""
    return np.fromiter(string, "U1")

def guessab(seq):
    """Guess sequence alphabet."""
    dna = ('A', 'T', 'C', 'G', 'Y', 'R', 'W', 'S', 'K', 'M', 'D', 'V', 'H', 'B', 'X', 'N')
    rna = ('A', 'U', 'C', 'G', 'Y', 'R', 'W', 'S', 'K', 'M', 'D', 'V', 'H', 'B', 'X', 'N')
    prot = np.array([
            'A', 'R', 'N', 'D', 'C', 'Q', 'E', 'G', 'H', 'I', 'L', 'K', 'M',
            'F', 'P', 'S', 'T', 'W', 'Y', 'V', 'B', 'Z', 'X', '*'], dtype='<U1')
    code = np.unique(seq)
    alphabet = 'NONE'
    if np.isin(code, dna).all():
        alphabet = 'DNA'
    elif np.isin(code, rna).all():
        alphabet = 'RNA'
    elif np.isin(code, prot).all():
        alphabet = 'PROT'
    return alphabet

def read_fasta(path, alphabet='guess'):
    """Load sequences from fasta. Return data structs."""
    dtype = np.dtype([
        ('id', (np.str_, 20)), ('name', (np.str_, 50)), ('desc', (np.str_, 120)),
        ('alphabet', (np.str_, 8)), ('start', int), ('end', int)])
    meta = []
    structs = []
    start = 0
    for data in SeqIO.parse(path, 'fasta'):
        end = start + len(data.seq)
        name = re.search('\A[^(,)]*', data.description.split('␣', 1)[-1]).group(0)
        seq = strtoseq(str(data.seq))
        alphabet = guessab(seq) if alphabet=='guess' else alphabet
        meta.append(np.array((data.id, name, data.description, alphabet, start, end), dtype=dtype))
        structs.extend(seqtorecs(seq, start))
        start = end
    return np.asarray(meta), structs
```

```
def get_codex(encoding):
    """Return codex dictionary for selected encoding."""
    defval = 0+0j
    if encoding == 'NA':
        alphabet = np.array([
            'A', 'T', 'U', 'C', 'G', 'Y', 'R', 'W',
            'S', 'K', 'M', 'D', 'V', 'H', 'B', 'X', 'N'], dtype='<U1')
        encoded = np.array([
            'A', 'T', 'T', 'C', 'G', 'Y', 'R', 'W',
            'S', 'K', 'M', 'D', 'V', 'H', 'B', 'N', 'N'], dtype='<U1')
        defval = 'N'
    elif encoding == 'COMPL':
        alphabet = np.array([
            'A', 'T', 'U', 'C', 'G', 'Y', 'R', 'W',
            'S', 'K', 'M', 'D', 'V', 'H', 'B', 'X', 'N'], dtype='<U1')
        encoded = np.array([
            'T', 'A', 'A', 'G', 'C', 'R', 'Y', 'W',
            'S', 'M', 'K', 'H', 'B', 'D', 'V', 'N', 'N'], dtype='<U1')
        defval = 'N'
    elif encoding == 'SQUARE':
        alphabet = np.array(['A', 'T', 'U', 'C', 'G'], dtype='<U1')
        encoded = np.array((1-1j, 1+1j, -1+1j, -1-1j), dtype=np.complex64)
    elif encoding == 'MAFFT':
        alphabet = np.array(['A', 'T', 'U', 'C', 'G'], dtype='<U1')
        encoded = np.array((1j, -1j, -1, 1), dtype=np.complex64)
    elif encoding=='AG':
        alphabet = np.array(['A', 'T', 'U', 'C', 'G'], dtype='<U1')
        encoded = np.array((1, 0, 0, 0, 1j), dtype=np.complex64)
    elif encoding=='PUPY':
        alphabet = np.array(['A', 'T', 'U', 'C', 'G'], dtype='<U1')
        encoded = np.array((1, 1j, 1j, 1j, 1), dtype=np.complex64)
    elif encoding == 'TRANS':
        alphabet = np.array(['A', 'T', 'C', 'G'], dtype='<U1')
        encoded = np.array([
            0.8944271 +0.4472137j , -0.4472138 -0.8944271j ,
           -0.8944272 -0.44721368j,  0.44721365+0.8944272j ], dtype=np.complex64)
    elif encoding == 'NUC.4.4':
        alphabet = np.array([
            'A', 'T', 'G', 'C', 'S', 'W', 'R', 'Y',
            'K', 'M', 'B', 'V', 'H', 'D', 'N'], dtype='<U1')
        encoded = np.array([
           -0.45467922-0.8906553j , -0.51913923+0.8546897j ,
            0.99873143-0.05035446j, -0.48378995+0.8751841j ,
            0.8763384 +0.48169592j, -0.8554102 -0.5179511j ,
            0.50102246-0.8654343j , -0.50232255+0.8646803j ,
            0.876649  +0.48113045j, -0.8551498 -0.51838094j,
            0.5562552 +0.83101153j,  0.51993096-0.8542083j ,
           -0.99772054-0.06748131j,  0.48131755-0.87654626j,
            0.49023774-0.87158877j], dtype=np.complex64)
    elif encoding=='BLOSUM62':
        alphabet = np.array([
            'A', 'R', 'N', 'D', 'C', 'Q', 'E', 'G', 'H', 'I', 'L', 'K', 'M',
            'F', 'P', 'S', 'T', 'W', 'Y', 'V', 'B', 'Z', 'X', '*'], dtype='<U1')
        encoded = np.array([
           -0.26695704-0.9637084j ,  0.61654115-0.7873227j ,
            0.89596575-0.44412315j,  0.9572207 -0.28935888j,
           -0.8933483 -0.4493649j ,  0.69961226-0.7145227j ,
            0.8500247 -0.5267428j ,  0.9853714 +0.17042053j,
            0.5077042 -0.86153144j, -0.93440825-0.35620385j,
           -0.93607223-0.35180783j,  0.6905609 -0.72327423j,
           -0.8275205 -0.5614355j , -0.99233466-0.12357972j,
            0.5013154 -0.8652646j ,  0.5061711 -0.8624331j ,
```

```
            -0.13414967-0.9909611j , -0.7313355 +0.68201786j,
            -0.9242165 -0.38186896j, -0.8691504 -0.49454784j,
             0.94218355-0.33509728j,  0.81015223-0.58621955j,
            -0.04461522-0.99900424j,  0.16116107+0.9869281j ], dtype=np.complex64)
    elif encoding=='PROPS':
        alphabet= np.array([
            'A', 'R', 'N', 'D', 'C', 'Q', 'E', 'G', 'H', 'I', 'L', 'K', 'M',
            'F', 'P', 'S', 'T', 'W', 'Y', 'V'], dtype='<U1')
        encoded = np.array([
            -0.85851073-0.51279557j,  0.21812595+0.9759206j ,
            -0.00420049+0.9999912j , -0.28015   +0.9599562j ,
            -0.65689373-0.7539832j ,  0.47489336+0.8800433j ,
            -0.01283448+0.9999176j , -0.978518  -0.20616122j,
             0.29292223+0.9561363j ,  0.07060943-0.99750406j,
             0.04576658-0.99895215j, -0.03847059+0.9992597j ,
             0.24712168-0.9689845j ,  0.40146118-0.91587603j,
            -0.7614468 +0.64822745j, -0.98953974+0.14426042j,
            -0.99279606+0.11981639j,  0.85201293-0.52352077j,
             0.99998915+0.00465518j, -0.18931031-0.9819173j ], dtype=complex64)
    codex = defaultdict(lambda: defval, zip(alphabet, encoded))
    return codex

def codexmap(data, codex):
    """Transform iterable using codex dictionary and return array."""
    if isinstance(codex, str):
        codex = get_codex(codex)
    uniq, inv = np.unique(data, return_inverse=True)
    return np.array([codex[u] for u in uniq])[inv].reshape(data.shape)

def seqtorecs(seq, start=0):
    end = start + seq.size
    locs = np.arange(start, end)
    dtype = np.dtype([
        ('loc', int), ('seq', (np.str_, 1)),
        ('enc', np.complex64), ('phase', bool)])
    seq = codexmap(seq, 'NA')
    revcomp = codexmap(seq[::-1], 'COMPL')
    recs = [np.empty(seq.size, dtype), np.empty(seq.size, dtype)]
    recs[0]['seq'] = seq
    recs[1]['seq'] = revcomp
    recs[0]['loc'] = locs
    recs[1]['loc'] = locs[::-1]
    recs[0]['phase'] = True
    recs[1]['phase'] = False
    return recs

def encode(meta, structs, encodings='SQUARE'):
    """Apply complex encoding to sequences."""
    alphabets = meta['alphabet'].repeat(2)
    encodings = np.atleast_1d(encodings)
    encodarr = []
    for encoding in encodings:
        if encoding in ('SQUARE', 'MAFFT', 'TRANS', 'AG', 'PUPY', 'NUC.4.4'):
            encodeds = []
            for struct in structs:
                encoded = codexmap(struct['seq'], encoding)
                copy = struct.copy()
                copy['enc'] = encoded
                encodeds.append(copy)
            encodarr.append(encodeds)
        elif encoding in ('BLOSUM62', 'PROPS'):
            encodeds = []
```

```
            for struct in structs:
                for frame in np.arange(3):
                    struct = struct[frame:]
                    struct = struct[:(struct.size//3)*3]
                    seq = np.ascontiguousarray(struct['seq'])
                    seq = seq.view(np.dtype((np.str_, 3)))
                    encoded = codexmap(seq, getcodons())
                    encoded = codexmap(encoded, get_codex('BLOSUM62'))
                    copy = struct.copy()
                    copy['enc'] = encoded.repeat(3)
                    encodeds.append(copy)
            encodarr.append(encodeds)
    return encodarr

def stft(encoded, window, detr='constant'):
    """Apply short-time fourier transform."""
    compl = np.iscomplexobj(encoded)
    if detr=='compcorr':
        detr = compcorr
    return signal.stft(encoded, 1, window, window.size, window.size-1,
                       return_onesided=not compl, boundary=None, axis=0, detrend=detr)[2]

def get_window(window_type, window_size, one_sided=True):
    """Get fourier tranform window."""
    if one_sided:
        window_size *= 2
    if window_type=='welch':
        window = np.arange(1,window_size+1)
        window = 1 - ((window-(window_size+1)/2)/((window_size+1)/2))**2
    else:
        window = signal.get_window(window_type, window_size, False)
    if one_sided:
        window = window[window_size//2-1::-1]
    return window / window.sum()

def norm_freqs(freqs, norm_level=1):
    """Normalise signal frequencies."""
    freqs = freqs * np.abs(freqs)**(norm_level-1)
    norm = np.linalg.norm(freqs, axis=-1, keepdims=True)
    norm[norm==0] = 1
    freqs /= norm
    return freqs

def compcorr(encoded, axis=0):
    """Correct for GC composition."""
    encoded -= encoded.mean(axis, keepdims=True)
    encoded.real /= np.abs(encoded.real).mean(axis, keepdims=True)*2
    encoded.imag /= np.abs(encoded.imag).mean(axis, keepdims=True)*2
    return encoded

def freq_transform(encoded, encoding='SQUARE', window_size=210, window_type='boxcar',
                   detr='constant', norm_level=1, chunk_size=24):
    """Apply frequency-space transformation."""
    gap = 1
    weight = 1
    if encoding in ('BLOSUM62', 'PROPS'):
        gap = 3
        weight = 1/4
    dtype = np.dtype([
        ('loc', int), ('seq', (np.str_, 1)), ('encs', (np.complex64, window_size)),
        ('freqs', (np.complex64, window_size)), ('powers', (float, window_size)),
        ('chunks', (np.float32, (window_size//chunk_size)*(chunk_size-1))),
```

```
        ('cwt', (np.complex64, window_size)),
        ('phase', bool), ('weight', float)])
    window = get_window(window_type, window_size)
    valid_size = encoded.shape[0] - window_size - gap
    freqs = stft(encoded['enc'], window, detr).swapaxes(0,1)[gap:valid_size+gap]
    powers = np.abs(freqs)
    encs = np.lib.stride_tricks.as_strided(encoded['enc'], (valid_size+gap,window_size),
                                           (encoded['enc'].strides[0],)*2)[gap:]
    encs = compcorr(encs, -1)
    n_chunk = window_size // chunk_size
    chunks = encs.reshape(valid_size, n_chunk, chunk_size)
    encs *= window.reshape(1,-1)
    chunks = np.abs(fft.fft(chunks, axis=-1)[:,:,1:])
    chunks /= np.linalg.norm(chunks, axis=-1, ord=2, keepdims=True)
    chunks = chunks.reshape(-1, (window_size//chunk_size)*(chunk_size-1))
    struct = np.empty(valid_size, dtype)
    struct['loc'] = encoded['loc'][:valid_size]
    struct['seq'] = encoded['seq'][:valid_size]
    struct['encs'] = encs
    struct['freqs'] = freqs
    struct['powers'] = powers
    struct['chunks'] = chunks
    struct['phase'] = encoded['phase'][:valid_size]
    struct['weight'] = weight
    return struct

def prepdata(encodeds, encodings='SQUARE', window_size=210, chunk_size=24, window_type='boxcar',
             detr='constant', norm_level=1):
    """Apply encodings to sequence structs."""
    encodings = np.atleast_1d(encodings)
    data = [np.concatenate(
        [freq_transform(enc, encoding, window_size, window_type, detr, norm_level, chunk_size)
        for enc in encod], 0)
            for encod, encoding in zip(encodeds, encodings)]
    return data

def get_data(datas, window_size=210, chunk_size=24, window_type='boxcar', encodings='SQUARE',
             detr='constant', norm_level=1):
    """Encode and normalise sequences, then pack into structs."""
    metalist, structlist, seqlist = [], [], []
    for data in datas:
        meta, seqs = data
        encods = encode(meta, seqs, encodings)
        structs = prepdata(encods, encodings, window_size, chunk_size, window_type, detr, norm_level)
        metalist.append(meta)
        structlist.append(structs)
        seqlist.append(seqs)
    return metalist, structlist, seqlist

def calcpm(seq1, seq2=None):
    """Caculate probability of bases matching due to chance."""
    if seq2 is None:
        seq1, seq2 = seq1[:,0], seq1[:,1]
    codex = {'A':0, 'T':1, 'U':1, 'C':2, 'G':3}
    probmatch = np.bincount(codexmap(seq1, codex), minlength=4) / seq1.size
    probmatch = probmatch * np.bincount(codexmap(seq2, codex), minlength=4) / seq2.size
    probmatch = probmatch.sum()
    probmatch
    return probmatch

def prepsearch(struct, guide_size=24, query=False):
    """Prepare search windows."""
```

```
    window_size = struct['encs'].shape[1]
    freqs = struct['encs'][:,:guide_size]
    freqs /= np.linalg.norm(freqs, axis=-1, ord=2, keepdims=True)
    freqs = np.concatenate((freqs.real, freqs.imag), axis=-1)
    powers = struct['chunks'][:,guide_size:]
    powers /= np.linalg.norm(powers, axis=-1, ord=2, keepdims=True)
    stack = np.concatenate((powers, freqs/3), axis=-1)
    stack /= np.linalg.norm(stack, axis=-1, ord=2, keepdims=True)
    return stack

def nnsearch(query, subject, guide_size=24, reduct='mip', index="Flat", n=1, sample='all'):
    """Run nearest neighbour search for a specified pair of sequences."""
    n = np.atleast_1d(n)
    dtype = np.dtype([
        ('locs', (int, 2)), ('seqs', ((np.str_, 1), 2)),
        ('score', float), ('phases', bool, 2), ('weight', float), ('query', float)])
    if reduct=='mip':
        sreduced = prepsearch(subject, guide_size)
        qreduced = prepsearch(query, guide_size, query=True)
    sampoints = np.arange(qreduced.shape[0])
    if isinstance(sample, float):
        if sample<1.0:
            sample = np.round(sample*qreduced.shape[0]).astype(int)
    if isinstance(sample, int):
        if sample<qreduced.shape[0]:
            sampoints = np.round(np.linspace(0, qreduced.shape[0]-1, sample, endpoint=True)).astype(int)
            qreduced = qreduced[sampoints]
    if isinstance(index, str):
        nn = faiss.index_factory(sreduced.shape[-1], index)
    else:
        nn = index
    if nn.is_trained == False:
        rng = default_rng()
        nn.train(rng.choice(sreduced, size=int(np.sqrt(sreduced.shape[0])), replace=False))
    nn.add(sreduced)
    _, matchlocs = nn.search(qreduced, int(n.max()))
    matchlocs = matchlocs[:,n-1].T.ravel()
    match = subject[matchlocs]
    query = np.ascontiguousarray(query[sampoints])
    query = np.lib.stride_tricks.as_strided(
        query, (n.size, query.size), (0, query.strides[0])).ravel()
    struct = np.empty(matchlocs.size, dtype)
    struct['locs'][:,0] = query['loc']
    struct['locs'][:,1] = match['loc']
    struct['seqs'][:,0] = query['seq']
    struct['seqs'][:,1] = match['seq']
    struct['score'] = np.real(query['freqs'] * match['freqs'].conj()).sum(1)
    struct['phases'] = np.stack([query['phase'], match['phase']], axis=1)
    struct['weight'] = (query['weight'] + match['weight'])/2

    order = np.lexsort([struct['score'], struct['locs'][:,0]])
    struct = struct[order]
    return struct

def nnsearches(queries, subjects, guide_size=24, reduct='mip', index="Flat", nmatch=1, sample='all'):
    """Run nearest-neighbour search for all sequence pairs."""
    structs = []
    for query, subject in zip(queries, subjects):
        structs.append(nnsearch(query, subject, guide_size, reduct, index, nmatch, sample))
    return structs

def multisearch(structlist, guide_size=24, reduct='mip', index="Flat", nmatch=1, sample='all'):
```

```
    """Run nearest-neighbour search."""
    nns = nnsearches(structlist[0], structlist[1], guide_size, reduct, index, nmatch, sample)
    for search in nns:
        search['query'] = True
    nns2 = nnsearches(structlist[1], structlist[0], guide_size, reduct, index, nmatch, sample)
    for search in nns2:
        search['locs'] = search['locs'][:,::-1]
        search['seqs'] = search['seqs'][:,::-1]
        search['phases'] = search['phases'][:,::-1]
        search['query'] = False
    nns.extend(nns2)
    nnarr = np.empty(len(nns), dtype=object)
    nnarr[:] = nns
    return nnarr

def get_windows(structs, seqlist, window_size):
    """Get window sequences."""
    phase1, phase2 = structs['phases'].T
    locs1, locs2 = structs['locs'].T
    seqs1 = np.stack([np.concatenate([seqs['seq'] for seqs in seqlist[0][::2]]), \
                      np.concatenate([seqs['seq'][::-1] for seqs in seqlist[0][1::2]])], axis=1)
    seqs2 = np.stack([np.concatenate([seqs['seq'] for seqs in seqlist[1][::2]]), \
                      np.concatenate([seqs['seq'][::-1] for seqs in seqlist[1][1::2]])], axis=1)

    winds1 = np.reshape(phase1*2-1, (-1,1)) * np.arange(1,window_size+1).reshape(1,-1)
    winds1 = locs1.reshape(-1,1) + winds1
    winds2 = np.reshape(phase2*2-1, (-1,1)) * np.arange(1,window_size+1).reshape(1,-1)
    winds2 = locs2.reshape(-1,1) + winds2
    seqs1 = np.squeeze(np.take_along_axis(seqs1[winds1], 1-1*phase1.reshape(-1,1,1), axis=2))
    seqs2 = np.squeeze(np.take_along_axis(seqs2[winds2], 1-1*phase2.reshape(-1,1,1), axis=2))
    return seqs1, seqs2

def filt_guides(seqs1, seqs2, order=None, max_pass=None, maxgaps=0, gapopen=0,
                match=1, gapext=0, mismatch=0, verbose=False):
    """Align and filter guide windows that have more than the specified number of gaps."""
    if order is not None:
        seqs1 = seqs1[order]
        seqs2 = seqs2[order]
    guide_size = seqs1.shape[1]
    seqs1 = seqs1.view(np.dtype((str, guide_size)))[:,0]
    seqs2 = seqs2.view(np.dtype((str, guide_size)))[:,0]
    submat = parasail.matrix_create("ATUCG", match, mismatch)
    sgfilt = np.zeros(seqs1.size, bool)

    if verbose:
        seqs1 = tqdm(seqs1)
    for i,(seq1,seq2) in enumerate(zip(seqs1,seqs2)):
        if max_pass is not None:
            if i==max_pass:
                break
        aln = parasail.sg_trace_scan_sat(seq1, seq2, gapopen, gapext, matrix=submat)
        query = np.array([aln.traceback.query]).view('U1')
        ref = np.array([aln.traceback.ref]).view('U1')
        ngaps = np.logical_or(query=='-',ref=='-').sum()
        sgfilt[i] = ngaps <= maxgaps
    if order is not None:
        order = np.argsort(order)
        sgfilt = sgfilt[order]
    return sgfilt

def alnseqs2(seqs1, seqs2, probmatch=0.25, gapopen=2, match=1, gapext=0, mismatch=0, verbose=False):
    """Align sequences."""
```

```
    window_size = seqs1.shape[1]
    seqs1 = seqs1.view(np.dtype((str, window_size)))[:,0]
    seqs2 = seqs2.view(np.dtype((str, window_size)))[:,0]
    submat = parasail.matrix_create("ATUCG", match, mismatch)
    totl = len(seqs1[0]) + len(seqs2[0])
    alns = ma.masked_array(np.full((seqs1.size,totl), np.uint8(2)), \
                           np.ones((seqs1.size,totl),bool), fill_value=np.uint8(2))

    if verbose:
        seqs1 = tqdm(seqs1)
    for i,(seq1,seq2) in enumerate(zip(seqs1,seqs2)):
        aln = parasail.sg_trace_scan_sat(seq1, seq2, gapopen, gapext, matrix=submat)
        query = np.array([aln.traceback.query]).view('U1')
        ref = np.array([aln.traceback.ref]).view('U1')
        gapq, gapr = query=='-', ref=='-'
        match = query==ref
        alns.data[i,:match.size] = match + gapq*2 + gapr*3
        alns.mask[i,:match.size] = False
    return alns


def stabil_binom(vals):
    """Apply arcsin stabilisation to binomially distributed values."""
    vals = np.arcsin((2*vals-1)/1) / np.pi * 2
    return vals

def destabil_binom(vals):
    """Inverse arcsin stabilisation."""
    vals = (np.sin(vals/2*np.pi)*1 + 1) / 2
    return vals

def calc_occs2(obs, width=None):
    """Count the numbers of k-mers."""
    length = obs.shape[0]
    if width is None:
        width = obs.max()+1
    occarr = obs + (np.arange(length)*width).reshape(-1,1)
    occarr = np.bincount(occarr.ravel(), minlength=length*width)
    occarr = occarr.reshape(length,width)
    return occarr

def kmer_occs(obs, mask=None, ksize=1, width=3):
    """Calculate the occurrence of alignment k-mers."""
    obs = as_strided(obs, (obs.shape[0],obs.shape[1]-ksize+1,ksize), \
                          (obs.strides[0],obs.strides[1],obs.strides[1]))
    obs = np.sum(obs * (obs.max()+1)**np.arange(ksize).reshape(1,1,-1), -1)
    if mask is not None:
        mask = as_strided(mask, (mask.shape[0],mask.shape[1]-ksize+1,ksize), \
                                (mask.strides[0],mask.strides[1],mask.strides[1]))
        mask = mask.any(-1)
        obs = (obs+1)*np.logical_not(mask)
    occarr = calc_occs2(obs, width+1)[:,1:]
    return occarr, obs

def calc_pindels(obs):
    """Calculate the corrected proportion of gaps in an alignment."""
    isgap = obs>=3
    isnuc = np.logical_or(obs==1, obs==2)
    isend = obs==0
    first = isgap[:,0]
    last = obs.shape[1] - np.argmin(isend[:,::-1],1) - 1
    last = isgap[np.arange(last.size),last]
```

```
    nindels = np.sum(np.logical_and(np.logical_not(isgap[:,:-1]), isgap[:,1:]), 1)
    nindels += np.logical_and(last==False, first==True)
    nindels[np.logical_and(nindels==0,first==True)] = 1
    nnucs = np.logical_and(isnuc[:,1:],isnuc[:,:-1]).sum(1)
    rawpinds = nindels / np.clip(nnucs+nindels, np.finfo(float).resolution, None)
    rawpinds = np.clip(rawpinds, 0, 0.5)
    pindels = 1 - (2-1/(1-rawpinds))
    return pindels

@tf.function
def clust_dists(in_poccs, comps, samp_vars, binpow=8., eps=1e-6):
    """Calculate distance from each point to nearest cluster."""
    diffs = tf.expand_dims(in_poccs, -3) - tf.expand_dims(comps, -2)
    dists = tf.sqrt(tf.reduce_sum(diffs**2/samp_vars[:,None,:], -1))
    simils = 1/tf.maximum(dists, eps)
    simils = simils/tf.norm(simils, axis=-1, keepdims=True)
    simils = simils**2
    clustw = simils*2 - 1
    clustw = tf.abs(clustw)**(1/binpow) * tf.sign(clustw)
    clustw = clustw/2 + 0.5
    return dists, simils, clustw

@tf.function
def score_poccs(in_poccs, comps, samp_vars, stabil_pm, learn_mult=1., binpow=8., eps=1e-6):
    """Calculate loss for current parameters."""
    in_poccs = 1/(1+tf.exp(-in_poccs))
    in_poccs = in_poccs*2 - 1
    stabil_pm = tf.reshape(stabil_pm, (1,)*len(in_poccs.shape))
    in_poccs = tf.concat([in_poccs, stabil_pm], -1)
    shape = tf.concat([in_poccs.shape[:-1], [in_poccs.shape[-1]//comps.shape[-1],comps.shape[-1]]], 0)
    in_poccs = tf.reshape(in_poccs, shape)

    dists, simils, clustw = clust_dists(in_poccs, comps, samp_vars, binpow, eps)
    loss = tf.reduce_sum(tf.reduce_sum(dists**2*clustw, -2) / (tf.reduce_sum(clustw, -2)-1))
    loss *= learn_mult
    return loss, in_poccs, dists, clustw

def calc_elliptw(xs, ys):
    """Apply elliptical distance metric from each point to cluster centers."""
    clustw = np.sqrt(xs**4 + 2*xs**2*(ys**2-1) + (ys**2+1)**2)
    clustw = np.sqrt(xs**2 + ys**2 + 1 - clustw)
    clustw = np.sign(xs) * clustw / np.sqrt(2)
    return clustw

def grad_desc2(comps, mean_preds, samp_vars, matches, pindels, probmatch=0.25, ndists=1,
               weights=None, binpow=16, reltol=1e-6, maxiter=100, learn_mult=0.01,
               prior_size=0, eps=1e-15, verbose=False):
    """Apply gradient descent algorithm to find homologous and non-homologous cluster centers."""
    test_poccs = comps[mean_preds>probmatch][np.argmin(np.abs(
        mean_preds[mean_preds>probmatch]-np.quantile(mean_preds[mean_preds>probmatch], 0.9)))]
    null_poccs = comps[np.argmin(np.abs(mean_preds-probmatch))]
    in_poccs = np.concatenate([test_poccs, null_poccs[:-1]])
    in_poccs = (np.clip(in_poccs, eps-1, 1-eps)+1)/2
    in_poccs = np.log(in_poccs/(1-in_poccs))

    stabil_pm = stabil_binom(probmatch)
    params = [comps, samp_vars, stabil_pm, learn_mult, binpow, eps]
    params = [np.float32(p) for p in params]
    func = lambda in_poccs: score_poccs(in_poccs, *params)[0]
    val_grad = lambda in_poccs: tfp.math.value_and_gradient(func, in_poccs)
    opt = tfp.optimizer.bfgs_minimize(
        val_grad,
```

```
        initial_position=tf.constant(in_poccs, tf.float32),
        tolerance=0,
        f_relative_tolerance=reltol,
        max_iterations=maxiter)

    pos = opt.position
    loss, out_poccs, cdists, clustw = score_poccs(pos, *params)
    loss, out_poccs, cdists, clustw = loss.numpy(), out_poccs.numpy(), cdists.numpy(), clustw.numpy()
    if verbose:
        print('niters: ' + str(opt.num_iterations.numpy()))
        print('loss: ' + str(opt.objective_value.numpy()))
        print('grad: ' + str(opt.objective_gradient.numpy()))

    diff = out_poccs[...,:-1,:] - out_poccs[...,-1:,:]
    bases = np.sqrt(np.sum(diff[None,:,:]**2/samp_vars[:,None,:], -1))
    sides = np.append(cdists, bases, -1)
    semipers = sides.sum(-1) / 2
    areas = np.sqrt(np.clip(semipers*np.prod(semipers[...,None]-sides, -1), 0, None))
    heights = 2 * areas[...,None] / bases
    projs = np.sqrt(cdists.max(-1,keepdims=True)**2-heights**2)
    mask = cdists.argmax(-1) != (cdists.shape[-1]-1)
    projs[mask] = bases[mask] - projs[mask]
    projs, heights = projs/bases*2-1, heights/bases
    clustw = calc_elliptw(projs, heights)
    clustw = clustw/2 + 0.5
    if binthres is None:
        clustw = clustw*2 - 1
        clustw = np.abs(clustw)**(1/binpow) * np.sign(clustw)
        clustw = clustw/2 + 0.5
    else:
        clustw = (clustw>binthres) * 1.
    clustw = np.append(clustw, np.max(1-clustw,-1,keepdims=True), -1)

    clust_sizes = np.clip(clustw.sum(0), 1, None)
    pids = []

    pid = None
    if out_poccs.shape[-1]>=2:
        out_poccs = destabil_binom(out_poccs)
        ngaps = out_poccs[...,-2] * clust_sizes * 1.5
        mtch = out_poccs[...,-1] * clust_sizes
        pid = (mtch+(prior_size-ngaps)*probmatch) / (clust_sizes+prior_size-ngaps)
        pids.append(norm_pid(pid.max(), probmatch))
    return clustw, pids

def norm_pid(percid, probmatch, lower=0.25, clip=False):
    """Normalise proportion identity to range 0-1."""
    pid = (percid-probmatch)/(1-probmatch)
    if clip:
        pid = np.clip(pid, 0, 1)
    pid = pid*(1-lower) + lower
    return pid

def JCdist(norm, clip=10, limit=0.25):
    """Convert identity to substitution distance via Jukes-Cantor model."""
    factor = 1 - limit
    mindist = np.e**(-1/factor*clip)
    norm = np.clip(norm, mindist, 1)
    return np.abs(-factor*np.log(norm))

    """."""
```

```
with StringIO(infile1.read()) as f:
    genome1 = read_fasta(f)
with StringIO(infile2.read()) as f:
    genome2 = read_fasta(f)
infile1.close()
infile2.close()

metalist, structlist, seqlist = \
    get_data([genome1, genome2], window_size, chunk_size, window_shape, encodings, detr, norm_level)

seq1, seq2 = seqlist[0][0]['seq'], seqlist[1][0]['seq']
probmatch = calcpm(seq1, seq2)

nns = multisearch(structlist, guide_size, reduct, index, nmatch, sample)
structs = np.concatenate(nns, 0)
_, mask = np.unique(structs, return_index=True)
structs = structs[mask]
coords = structs['locs']
seqs = structs['seqs']
matches = seqs[:,0]==seqs[:,1]
concord = structs['phases'][:,0]==structs['phases'][:,1]

seqs1, seqs2 = get_windows(structs, seqlist, window_size)
guides1 = np.ascontiguousarray(seqs1[:,:filt_size])
guides2 =  np.ascontiguousarray(seqs2[:,:filt_size])
sgfilt = filt_guides(guides1, guides2, verbose=verbose)

order = np.argsort(-structs['score'])
rev = np.argsort(order)
sgfilt = sgfilt[order]
opts = np.nonzero(sgfilt)[0]
if opts.size > max_pass:
    pass_filt = np.array([arr[0] for arr in np.array_split(opts, max_pass-1)])
    pass_filt = np.append(pass_filt, opts[-1])
    sgfilt[:] = False
    sgfilt[pass_filt] = True
sgfilt = sgfilt[rev]

structs = structs[sgfilt]
coords = structs['locs']
seqs = structs['seqs']
probmatch = calcpm(seqs)
matches = seqs[:,0]==seqs[:,1]
concord = structs['phases'][:,0]==structs['phases'][:,1]

seqs1, seqs2 = get_windows(structs, seqlist, window_size)
alns = alnseqs2(seqs1, seqs2, probmatch, 2, 1, verbose=verbose)
nsyms, sym_obs = kmer_occs(np.clip(alns.data, 0, 2), alns.mask, 1)

shape = (sym_obs.shape[0], sym_obs.shape[1]-samp_width+1, samp_width)
strides = (sym_obs.strides[0], sym_obs.strides[1], sym_obs.strides[1])
samp_obs = as_strided(sym_obs, shape, strides)
samp_pinds = calc_pindels(samp_obs.reshape(-1,samp_width)).reshape(samp_obs.shape[:-1])
samp_occs = calc_occs2(samp_obs.reshape(-1,samp_width))
samp_occs = samp_occs[:,1:].reshape(samp_obs.shape[0], samp_obs.shape[1], -1)
samp_poccs = samp_occs/np.clip(samp_occs.sum(-1,keepdims=True), 1, None)
samp_pmatchs = samp_poccs[...,1]

samp_vals = samp_pmatchs[...,np.newaxis]
samp_vals = stabil_binom(samp_vals)
samp_diffs = np.sqrt(np.sum((samp_vals-samp_vals[:,:1,:])**2*samp_width, -1))
samp_diffs[np.any(samp_obs==0, axis=-1)] = np.inf
samp_cuts = np.argmax(samp_diffs>cut_thres, 1) - 1
samp_cuts = np.clip(samp_cuts-samp_width+1, 2, None)
```

```
cutfilt = samp_cuts>=min_samps
structs = structs[cutfilt]
coords = structs['locs']
seqs = structs['seqs']
probmatch = calcpm(seqs)
matches = seqs[:,0]==seqs[:,1]
concord = structs['phases'][:,0]==structs['phases'][:,1]
alns = alns[cutfilt]
samp_vars = (samp_obs, samp_pinds, samp_pmatchs, samp_diffs, samp_cuts)
samp_obs, samp_pinds, samp_pmatchs, samp_diffs, samp_cuts = [samp_var[cutfilt] for samp_var in samp_vars]

samp_matchs = as_strided(matches, (samp_obs.shape[0],samp_obs.shape[1]), (matches.strides[0],0))
samp_cids = np.arange(samp_obs.shape[0])
samp_cids = as_strided(samp_cids, (samp_obs.shape[0],samp_obs.shape[1]), (samp_cids.strides[0],0))
samp_mask = np.zeros((samp_obs.shape[0], samp_obs.shape[1]), bool)
samp_mask[np.arange(samp_cuts.size), samp_cuts] = True
samp_mask = np.maximum.accumulate(samp_mask[:,::-1], 1)[:,::-1]
samp_mask *= np.all(samp_obs!=0, axis=-1)

samp_inv = np.stack(np.nonzero(samp_mask), 1)
samp_near = np.nonzero(samp_inv[:,1]==0)[0]
samp_obs, samp_matchs, samp_cids, samp_pinds, samp_pmatchs, samp_diffs = \
    samp_obs[samp_mask], samp_matchs[samp_mask], samp_cids[samp_mask], \
    samp_pinds[samp_mask], samp_pmatchs[samp_mask], samp_diffs[samp_mask]

alns.mask = np.concatenate([np.full((samp_mask.shape[0],samp_width-1),False), np.logical_not(samp_mask)], -1)
nsyms, sym_obs = kmer_occs(np.clip(alns.data, 0, 2), alns.mask, 1)
nmismats, nmatchs, ngaps = nsyms.T
pmatchs = nmatchs / nsyms.sum(1)
pmismats = nmismats / nsyms.sum(1)
pgaps = ngaps / nsyms.sum(1)
pindels = calc_pindels(sym_obs)

vals = np.stack([samp_pmatchs, samp_pinds], 1)
reg = lgb.LGBMRegressor(num_leaves=nleaves, max_depth=-1, mc=(1,-1), mc_method='advanced', \
                        n_jobs=ncpu, n_estimators=ntrees, learning_rate=learn_rate, \
                        subsample=subsamp, subsample_freq=1*(subsamp<1))
reg.fit(vals, samp_matchs.ravel())
samp_preds = reg.predict(vals)
samp_preds = np.clip(samp_preds, 0, 1)

predarr = np.full(samp_mask.shape, -1.)
predarr[samp_inv[:,0], samp_inv[:,1]] = stabil_binom(samp_preds)
mean_preds = np.average(predarr, weights=samp_mask, axis=1)
std_preds = np.sqrt(np.average((predarr-mean_preds[:,np.newaxis])**2, weights=samp_mask, axis=1))
predarr = destabil_binom(predarr)
mean_preds = destabil_binom(mean_preds)

comps = np.stack([samp_pmatchs, samp_pinds, samp_preds], 1)
comps = stabil_binom(comps)
samp_vars = np.stack([stats.binned_statistic(samp_cids, comps[:,c], 'std',
                                             np.arange(matches.size+1)).statistic**2 \
                      for c in np.arange(comps.shape[1])], -1)
comps = np.stack([samp_pmatchs, samp_pinds, np.clip(samp_preds, probmatch, None)], 1)
comps = stabil_binom(comps)
nobs = samp_mask.sum(1)
minvar = np.finfo(np.float32).resolution
samp_vars = np.clip(samp_vars, minvar, None)
samp_vars = samp_vars[samp_cids]

pred_mask = np.ones(samp_preds.shape, bool)
comps = comps[pred_mask]
```

```
samp_vars = samp_vars[pred_mask]

clustw, pids = grad_desc2(comps, samp_preds[pred_mask], samp_vars, matches, pindels,
                          probmatch=probmatch, ndists=1, weights=None, \
               binpow=binpow, reltol=reltol, maxiter=maxiter, learn_mult=learn_mult,
                          prior_size=prior_size, eps=eps, verbose=verbose)
clustw = np.stack([stats.binned_statistic(samp_cids[pred_mask], clustw[:,c], 'mean',
                                          np.arange(matches.size+1)).statistic \
               for c in np.arange(clustw.shape[-1])], -1)
clust_sizes = clustw.sum(0)

mtch = matches.reshape(-1,1)
pgs = pindels.reshape(-1,1)
ngaps = np.sum(mtch*pgs, 0)
pid = (mtch.sum(0)+(prior_size-ngaps)*probmatch) / (clust_sizes+prior_size-ngaps)
pids.append(norm_pid(pid.max(), probmatch))

mtch = mean_preds.reshape(-1,1) * clustw
pgs = pindels.reshape(-1,1) * clustw
ngaps = np.sum(mtch*pgs, 0)
pid = (mtch.sum(0)+(prior_size-ngaps)*probmatch) / (clust_sizes+prior_size-ngaps)
pids.append(norm_pid(pid.max(), probmatch))

mtch = matches.reshape(-1,1) * clustw
pgs = pindels.reshape(-1,1) * clustw
ngaps = np.sum(mtch*pgs, 0)
pid = (mtch.sum(0)+(prior_size-ngaps)*probmatch) / (clust_sizes+prior_size-ngaps)
pids.append(norm_pid(pid.max(), probmatch))

pids = np.array(pids)
dists = JCdist(norm_pid(pids, 0.25, 0, True))

out_dist = np.maximum(dists[1], dists[-1])

outfile.write(str(out_dist))
outfile.close()
```

Appendix C. Concatenated Rfam genomes

>0.0-0
UGCCAUCGUCGUCGAUCUCUAUCAACUACCCUUGCGACUAUGGCAGCAAUCCGGCCUUUAAACUACCAGCGGGCGGGCUCCCUACGGGGCAGUCCGCCCUUGGCCAACUU
AUCGAGUACGCGGGGGAAUUUAAUUCGCUAUCGAUUUGAGAAUGGGUAUGGCAAAGUAAGAAUAGAAAGCGCUUAUGUGACCAAACAGAGAAGUCAACCAGAGAAACACA
CGUUGUGGUAUAUUACCUGGUAGGGUAACCACUAAAAUCCCGAAAGGGUGGGCUGUGGUGACCUUCCUAGGGCCGUCAGGGUUCUCCCGCCCUCCGUGAACCCAGGCUAA
AAGUUAAGGUAGGGGGGCAUGGGCGGCGCAUGAGAGAAGCCCAAACCAAUAACUACCCAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGAUAGUCCCUUCCUCAGAG
CAUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACAGAUAAUGACCAUGCUAACGCCAGAGCGUUUUCGCAUUUGGCAUCGAAAUUGAUCGAGACG
GAGGUGGAACCAUCCGAUACGAUCCUAGACAUUGGAAGUGCGCCUGCCCGCAGAAUUUUGAAAGGGGUCUCCUAGAGAGCUUGCCUAGGGCCUUAUAACCCGACUUGCUG
AGCUUCUAUAGGAAAAAACCCUUUCCCAGCCUUGGGGUGGCUGGUCAAUAAAAACCCCCAUCAGGGUUCUUAAAUCUGUUUCUCUAUAAGAACACUCAUUUCACGCUUUC
UGUCUUCUUUCUUCCAGGGCUCUCCCCUUGCCCUAGGCUCUGGCCGUUGCGCCCGGCGGGGUCAACUCCAUGAUUAGCAUGGAGCUGUAGGAGUCUAAAUUGGGGACACA
GAUGUUUGGAACGUCACCUUGCAGUGUUAACUUGGCUUUCAUGAAUCUCUUUGAUCUUCCACAAGGGGUAGGCUACGGGUGAAACCUCUUAGGCUAAUACUUCUAUGAAG
AGAUGCCUUGGAUAGGGUAACAGCGGCGGAUAUUGGUGAGUUGUUAAGACAAAAACCAUUCAACGCCGGAGGACUGACUCUCAUCCAGUGGAUGCAUUGAGUGGAUUGAC
UGUCGGGGCUGUCUUUAGGCUUAAUUCCAGACCUCUCUGUGCUUGGGGCAAACAUCAUUUGGCCUUAAAUGGGAUUCUGUGAGAGGGGAUCCCUCCAUUGCCAGCUGGAC
UGUUCUUUGGGGCCUUAUGUGGUGUUUGCCGCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCAUUCUUAAAUAAUAAUGCGUGGUGCAUACGAUAAUGCAGGCGACGG
UGAGUACGCCAUAGCCUCCCUCCUCGCGCGGGGGGGGGGCCUGCAUGGGAAAUUCCCCCUCCUAUAACCCCCCCGCUGGUAUCUCCCCCUCAGACUGGCUGAUUUACCCC
AUUAAUUUGGGAUGCUAAAGUCAUUUAAUGCUGACCUCCACUGGGUGGAUUAAGGUCAAGGUAUGAAGUCCUAUUCGCUCCUGAUAGGAUCGACUUCAUAUUGCUUAUAU
AUGUGCUAACGCACAUAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGCCCCUAAGGGAUGCCUGGGAAAGACCAGAGAUCCUGCUGUCUCCUCAGCAUCAUUCCAGGCAC
AGAACGCCAGAAAAUGGAGGCUUAGGUGAUGAAUUAGAAGAAGUCAUCGUUGAGAAAACGAAACAGACGGUGGCCUCAAUCUCAUCUGGUCCAAAACACACACAGCCUCC
AGGCCUUGCAACACUUGGUCCGGAAGGCCCUGGAGGCGUUUUCAUCUUACACACGCUUGUGCAAGAUAGUUACGGUUUUCCUAAAUCGUCGUGAGACGGACCUUGGAAAA
CCUAGUAUUUAAAUAACUAGGUCGGGCCUUCUGGUCGUGUUCUUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGACAGCCAAGAGGUCUCCGAGGAGGUCGGGGUUGCC
CACCGUUUGAUCUACAACGGUGCGAAGCUCCGUACUUACACGCGGGCUCGUGGUCUUCACGAUAGAAGAAGUGUUUUUCCCUCCACUUAAAUCGAAGGGUGUUGGUUUUU
CUCCAAGGAUUUACUGACAACCGAAGCCCUGACUGUUACCGCAGUUGGGGCUUCAACUCUGUGUAGUGCCUUUUUUAGUCGUGGCCGGCGCCGCAGGCGCCGGCGCCUAC
GAGCACGCGACCACGAUGCCUCCUAGCGAAGUCCCGCUAGGAAGAAAAAAUAAGGGCCGAGUAGGACUAGCAAACGGAGGGACUAGCCGUAGUGGCGAGCUCCCUGGGUG
GUCUAAGUCCUGAGUACAGGACAGUCGUCAGUAGUUCGACGUGAGCAGAAGCCCACCUCGAGAUGCUAUGUGGACGAGGGCAUGCCCAAGACACACCUUAACCCUAGCGG
GGGUCGCUAGGGUGAAAUCACACCACGUGAUGGGAGUACGACCUGAUAGGGCGCUGCAGAGGCCCACUAUUAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGA
GUUCGUGGCCAUCACGAUAGAUGGUAAGUUCUCGAUCUUUAAAAUCGGAAUUCAUCGAUUCGGUUGCAGCAUUUAAAGCGGUUGACAAUUUCAGGAAAAAGAAAAAGAAG
AUUGGAGGAAGGGAUGUGAUUAGUAAGUAUAAGUAUAGACCAGAGAAGUACGCCGGUCCUGAUUCGUUACUUAAUAAAGAAGAAAAUGACACUACACUAUUUAUUUAUUUA
UUUCUUAGUAAUGUUUACUGGAACUUAAAUCUGUGAUAACCUAAACUAAAAGUCUCAAACAACCCGUUACUUGCUGACCAAUUUUUGUAGGUACUGGGUGUAAAUAUGAU
GACCGGUACCAUCUAACGGGAAGUGUGUAAGCACACACGUAAUUUGCAAGCGGUGCUUCACGCUCUUCGUUAAAAUUUGCUAUGCCGCGGCAGACUCCUUUUCCCGCCAA
GAACUUAUAGACCAGGAGAAAGAACUCCUUGAGAAGUUGGCGUGGCGAACAGAGGCAGUCUUAGCGACGGACGUCACUUCCUUCUUGUUACUUAAAUUGCUGGGGGGCUC
CCAACACCUGGACUUUUGGCACCACGAGGUCAACACCCUGAUUACAAAAGCCUUAGUUGACCCAAAGACUGGCUCAUUGCCCGCCUCUAUUAUCAGCGCUGCAGUACUUG
CGCGGCGCUAUAGCGCCUCGCUAGUGGGCCCCCGCGGAAACCCGCACUAGCCCACUAGAUUGAGCCCCUGACUGGGUUAAAGUCACAAUGAACCAACGAAAAAAGGUGGU
UAGACCACCUUUCAAUAUGGCUGUGGGCCGGCUUUUUUACUUGUCUGCGGCGCCUUGGGCGCCGCAGCGUACGAACACACAGCAGUGAUGCCAAAACAGCUCUGGGGUUG
CUCCCACCCCAGACCACGUGGCGGCAGUACUCUGGUAUUGCGGUACCUUUGUACGCCUGUUUUAUACAUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGC
AACAUUCCGAAGGGGACCGUCCCUCGGUAAUGGCGAAUGGGACCCAUGCCAAGGUAUCUUUACGUCUACAUUGCUGUUGUCGUGUGUGACUGUACCUUUGGUAUGACUAG
CGGAGGCUAGAAGGAGUGUUCAUGUCCUACUGUUCAAGCCUCCAAGCUGUGCCUUGGGUGGCUUUGGGGCAUGGACAAACUCAGGCCUAAACUCAUGCAGACCACACAAG
GCAGAUGGGCUAUAUAAACGUUUUCGCUUUUCCGUUUACGAUAUAUAGUCUACUCUUGUGCAGAAUGAAUUCUCGUAACUACAUAGCACAAGUAGAUGUAGUUAACUUUA
AUCUCACAUAGCAAUCUUUAAUCAGUGUGUAACAUUAGGGAGGACUUGAAAGAGCCACCACAUUUUCACCGAGGCCACGCGGAGUACGAUCGAGUGUACAGUGAACAAUG
CUAGGGAGAGCUGCCUAUAUGGAAGAGCCCUAAUGUGUAAAAUUAAUUUUAGUAGUGCUAUCCCCAUGUGAUUUUAAUAGCUUCUUAGGAGAAUGACCGUUUGGUCGAAU
GCUACUAGAGCAGCCUGACCAAACUAGUAUUAGUAAAUAAAGGAGAGUUAUCAUACUCUAUAAACUAUGAGCCGCUGGGAUUGCGGAUUAUAAAUCGCAGUGCCGAGCCA
CGCGGGGUACGAUCGAGGGUACAGCACUAUGGGUGCGAGAGCGUCAGUAGGUCUCUCUGUUAGACCAGAUCUGAGCCUGAGCUCUCUGCUAACUAGGAACCACUUUUUUA
GGGAAAAUCUGGCCUUCCCACAAGGGGAGGCCAGGAAUUCUCUUGGUAGCUAGAGAUCCCUUAGACCCUUUUAGUCAGUGUGGAAAAUCUCUAGCAGUGGCGCCCGAACA
GGGACUUGAAAGUGAAAGAACCAGAGACGGAUUGUGUCCGUAAUCACAAAUGGUACGGUACUUCCAUUGUCAUGUGUAUGUUGGGGAUUAAACCCUGAUUGAGUUCAGCC
CACAUACUUUGUUGAUUGGUUGUCAAUCAUGGCAAAAGUGCACGCUACUUUGAUAACCUUUUACAUGCGACACAGACGAAGCGCUAAAACGUGGGAUUCUGUGUCGUUUU
AUGUUGUUCAUUGACAAACCUAUCUAGUAGGUCUAUAGAUUGAUAUAUUAAGUAGUAGUUUCCCUCCCUGAUGGACUCGGCUUGUGAAGCGCGCACGGCAAGAGGCGAGG
GAUGUUAUCUUCAGCCGUACAGGGAGCCUUGAACCGAGCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAUGAAGCUCUAAG
CGAGUAAGACAGACUCUUCAGUCUGAGUUUGUGGAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUAAACGUUGCAUGAUCGUAAAACAUGCCCGGCUUACCGUGAGCC
GCCUUUGAAAGAAAGGUUUACACGGACUCUGUUGCGUUACGCACGUACCUAACCCGUGCCAGCAGAAUAAAGACCUUUUGGUUUCUACAGUUUUUUCC
>0.0-1
UGCCAUCGUCGUCGAUCUCUAUCAACUACCCUUGCGACUAUGGCAGCAAUCCGGCCUUUAAACUACCAGCGGGCGGGCUCCCUACGGGGCAGUCCGCCCUUGGCCAACUU
AUCGAGUACGCGGGGGAAUUUAAUUCGCUAUCGAUUUGAGAAUGGGUAUGGCAAAGUAAGAAUAGAAAGCGCUUAUGUGACCAAACAGAGAAGUCAACCAGAGAAACACA
CGUUGUGGUAUAUUACCUGGUAGGGUAACCACUAAAAUCCCGAAAGGGUGGGCUGUGGUGACCUUCCUAGGGCCGUCAGGGUUCUCCCGCCCUCCGUGAACCCAGGCUAA

AAGUUAAGGUAGGGGGGCAUGGGCGGCGCAUGAGAGAAGCCCAAACCAAUAACUACCCAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGAUAGUCCCUUCCUCAGAG
CAUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACAGAUAAUGACCAUGCUAACGCCAGAGCGUUUUCGCAUUUGGCAUCGAAAUUGAUCGAGACG
GAGGUGGAACCAUCCGAUACGAUCCUAGACAUUGGAAGUGCGCCUGCCCGCAGAAUUUUGAAAGGGGUCUCCUAGAGAGCUUGCCUAGGGCCUUAUAACCCGACUUGCUG
AGCUUCUAUAGGAAAAAACCCUUUCCCAGCCUUGGGGUGGCUGGUCAAUAAAAACCCCCAUCAGGGUUCUUAAAUCUGUUUCUCUAUAAGAACACUCAUUUCACGCUUUC
UGUCUUCUUUCUUCCAGGGCUCUCCCCUUGCCCUAGGCUCUGGCCGUUGCGCCCGGCGGGGUCAACUCCAUGAUUAGCAUGGAGCUGUAGGAGUCUAAAUUGGGGACACA
GAUGUUUGGAACGUCACCUUGCAGUGUUAACUUGGCUUUCAUGAAUCUCUUUGAUCUUCCACAAGGGUAGGCUACGGGUGAAACCUCUUAGGCUAAUACUUCUAUGAAG
AGAUGCCUUGGAUAGGGUAACAGCGGCGGAUAUUGGUGAGUUGUUAAGACAAAAACCAUUCAACGCCGGAGGACUGACUCUCAUCCAGUGGAUGCAUUGAGUGGAUUGAC
UGUCGGGGCUGUCUUUAGGCUUAAUUCCAGACCUCUCUGUGCUUGGGGCAAACAUCAUUUGGCCUUAAAUGGGAUUCUGUGAGAGGGGAUCCCUCCAUUGCCAGCUGGAC
UGUUCUUUGGGGCCUUAUGUGGUGUUUGCCGCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCAUUCUUAAAUAAUAAUGCGUGGUGCAUACGAUAAUGCAGGCGACGG
UGAGUACGCCAUAGCCUCCCUCCUCGCGCGGGGGGGGGGCCUGCAUGGGAAAUUCCCCCUCCUAUAACCCCCCCGCUGGUAUCUCCCCCUCAGACUGGCUGAUUUACCCC
AUUAAUUUGGGAUGCUAAAGUCAUUUAAUGCUGACCUCCACUGGGUGGAUUAAGGUCAAGGUAUGAAGUCCUAUUCGCUCCUGAUAGGAUCGACUUCAUAUUGCUUAUAC
UAUGUAACGCACAUAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGCCCCUAAGGGAUGCCUGGGAAAGACCAGAGAUCCUGCUGUCUCCUCAGCAUCAUUCCAGGCACAG
AACGCCAGAAAAUGGAGGCUUAGGUGAUGAAUUAGAAGAAGUCAUCGUUGAGAAAACGAAACAGACGGUGGCCUCAAUCUCAUCUGGUCCAAAACACACACAGCCUCCAG
GCCUUGCAACACUUGGUCCGGAAGGCCCUGGAGGCGUUUUCAUCUUACACACGCUUGUGCAAGAUAGUUACGGUUUUCCUAAAUCGUCGUGAGACGGACCUUGGAAAACC
UAGUAUUUAAAUAACUAGGUCGGGCCUUCUGGUCGUGUUCUUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGACAGCCAAGAGGUCUCCGAGGAGGUCGGGGUUGCCCA
CCGUUUGAUACUACAACGGUGCGAAGCUCCGUACUUACACGCGGGCUCGUGGUCUUCACGAUAGAAGAAGUGUUUUCCCUCCACUUAAAUCGAAGGGGUUGGUUUUUCUC
CAAGGAUUUACUGACAACCGAAGCCCUGACUGUUACCGCAGUUGGGGCUUCAACUCUGUGUAGUGCCUUUUUUAGUCGUGGCCGGCGCCGCAGGCGCCGGCGCCUACGAG
CACGCGACCACGAUGCCUCCUAGCGAAGUCCCGCUAGGAAGAAAAAAUAAGGGCCGAGUAGGACUAGCAAACGGAGGGACUAGCCGUAGUGGCGAGCUCCCUGGGUGGUC
UAAGUCCUGAGUACAGGACAGUCGUCAGUAGUUCGACGUGAGCAGAAGCCCACCUCGAGAUGCUAUGUGGACGAGGGCAUGCCCAAGACACACCUUAACCCUAGCGGGGG
UCGCUAGGGUGAAAUCACACCACGUGAUGGGAGUACGACCUGAUAGGGCGCUGCAGAGGCCCACUAUUAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUU
CGUGGCCAUCACGAUAGAUGGUAAGUUCUCGAUCUUUAAAAUCGGAAUUCAUCGAUUCGGUUGCAGCAUUUAAAGCGGUUGACAAUUUCAGGAAAAAGAAAAAGAAGAUU
GGAGGAAGGGAUGUGAUUAGUAAGUAUAAGUAUAGACCAGAGAAGUACGCCGGUCCUGAUUCGUUACUUAAUAAAGAAGAAAAUGACACUACACUAUUUUAUUUAUUUAUUU
CUUAGUAAUGUUUACUGGAACUUAAAUCUGUGAUAACCUAAACUAAAAGUCUCAAACAACCCGUUACUUGCUGACCAAUUUUUGUAGGUACUGGGUGUAAAUAUGAUGAC
CGGUACCAUCUAACGGGAAGUGUGUAAGCACACACGUAAUUUGCAAGCGGUGCUUCACGCUCUUCGUUAAAAUUUGCUAUGCCGCGGCAGACUCCUUUUCCCGCCAAGAA
CUUAUAGACCAGGAGAAAGAACUCCUUGAGAAGUUGGCGUGGCGAACAGAGGCAGUCUUAGCGACGGACGUCACUUCCUUCUUGUUACUUAAAUUGCUGGGGGGCUCCCA
ACACCUGGACUUUUGGCACCACGAGGUCAACACCCUGAUUACAAAAGCCUUAGUUGACCCAAAGACUGGCUCAUUGCCCGCCUCUAUUAUCAGCGCUGCAGUACUUGCGC
GGCGCUAUAGCGCCUCGCUAGUGGGCCCCCGCGGAAACCCGCACUAGCCCACUAGAUUGAGCCCCUGACUGGGUUAAAGUCACAAUGAACCAACGAAAAAAGGUGGUUAG
ACCACCUUUCAAUAUGGCUGUGGGCCGGCUUUUUUACUUGUCUGCGGCGCCUUGGGCGCCGCAGCGUACGAACACACAGCAGUGAUGCCAAAACAGCUCUGGGGUUGCUC
CCACCCCAGACCACGUGGCGGCAGUACUCUGGUAUUGCGGUACCUUUGUACGCCUGUUUUAUACAUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAAC
AUUCCGAGGGGACCGUCCCCUCGGUAAUGGCGAAUGGGACCCAUGCCAAGGUAUCUUUACGUCUACAUUGCUGUUGUCGUGUGUGACUGUACCUUUGGUAUGACUAGCGG
AGGCUAGAAGGAGUGUUCAUGUCCUACUGUUCAAGCCUCCAAGCUGUGCCUUGGGUGGCUUUGGGGCAUGGACAGACCACACAAGGCAGAUGGGCUAUAUAAACGUUUUC
GCUUUUCCGUUUACGAUAUAUAGUCUACUCUUGUGCAGAAUGAAUUCUCGUAACUACAUAGCACAAGUAGAUGUAGUUAACUUUAAUCUCACAUAGCAAUCUUUAAUCAG
UGUGUAACAUUAGGGAGGACUUGAAAGAGCCACCACAUUUUCACCGAGGCCACGCGGAGUAUACAGUGAACAAUGCUAGGGAGAGCUGCCUAUAUGGAAGAGCCCUAAUG
UGUAAAAUUAAUUUUAGUAGUGCUAUCCCCAUGUGAUUUUAAUAGCUUCUUAGGAGAAUGACCGUUUGGUCGAAUGCUACUAGAGCAGCCUGACCAAACUAGUAUUAGUA
AAUAAAGGAGAGUUAUCAUACUCUAUAAACUAUGAGCCGCUGGGAUUGCGGAUUAUAAAUCGCAGUGCCGAGCCACGCGGGGUACGAUCGAGGGUACAGCACUAUGGGUG
CGAGAGCGUCAGUAGGUCUCUCUGUUAGACCAGAUCUGAGCCUGAGCUCUCUGCUAACUAGGAACCACUUUUAGGGAAAAUCUGGCCUUCCCACAAGGGGAGGCCAGGAA
UUCUCUUGGUAGCUAGAGAUCCCUUAGACCCUUUUAGUCAGUGUGGAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACUUGAAAGUGAAAGAACCAGAGACGGAUUGUGU
CCGUAAUCACAAAUGGUACGGUACUUCCAUUGUCAUGUGUAUGUUGGGGAUUAAACCCUGAUUGAGUUCAGCCCACAUACUUUGUUGAUUGGUUGUCAAUCAUGGCAAAA
GUGCACGCUACUUUGAUAACCUUUUACAUGCGACACAGACGAAGCGCUAAAACGUGGGAUUCUGUGUCGUUUUAUGUUGUUCAUUGACAAACCUAUCUAGUAGGUCUAUA
GAUUGAUAUAUUAAGUAGUAGUUUCCCUCCCUGAUGGACUCGGCUUGUGAAGCGCGCACGGCAAGAGGCGAGGGAUGUUAUCUUCAGCCGUACAGGGAGCCUUGAACCGA
GCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAUGAAGCUCUAAGCGAGUAAGACAGACUCUUCAGUCUGAGUUUGUGGAGA
UGAGUGUAAAUCUGGCAUAGCAUACAGGUUAAACGUUGCAUGAUCGUAAAACAUGCCCGGCUUACCGUGAGCCGCCUUUGAAAGAAAGGUUUACACGGACUCUGUUGCGU
UACGCACGUACCUAACCCGUGCCAGCAGAAUAAAGACCUUUUGGUUUCUACAGUUUUUUCC
>0.05-0
GUCGGUACUAGUUAGCUAACUAGCUCUGUAUCUGGCGGACCCGUGGUGGAACUGACGAGUUCGGAACACCCGGCCGCGACCCUGGGAGACGUCCCAGGGACACUAAGGCA
AAGAGAAGAGUGGUGCAGAGAGAAAAAAGAGCAGUGGGAAUAGGAGCUGUACUCCUUGGGUUCUUGGGAGCAGCAGGAAGCACUAUGGGCGCAGCGUCAUUGACGCUGAC
GGUACAGGCCAGACUAUUAUUGUCUGGUAUAGUGCAACAGCAGAACAAUUUGCUGAGGGCUAUUGAGGCGCAACAGCAUCUGUUGCAACUCACAGUCUGGGGCAUCAAGC
AGCUCCAGGCAAGAGUCCUGGCUGUGGAAAGAUACCUAAAGGAUCAACAGCUCCUGGGAAUUUGGGGCUGCUCUGGAAAACUCAUUUGCACCACUAACAUGACCUGGAUG
GCCGGCAUGGUCCCAGCCUCCUCGUCGGCGCCGGCCGGGCAACAUUCCGAGGGGACCGUCCCUCGGUAAUGGCGAAUGGGACCCAAUGAAUAACCAACGAAAAAAGGCGA
AAAAUACGCCUUUCAAUAUGAAAACAGCUCUGGGGUUGCUCCCACCCCAGACCACGUGGCGGCAGUACUCUGGUAUGCGGUACCUUUGUACGCCUGUUUUAUACAGUAGG
ACUAGCAAAACGGAGGGACUAGCCAUAGUGGCGAGCUCCCUGGGUGGUCUAAGUCCUGAGUACAGGACAGUCGUCAGUAGUUCGACGUGAGCAGAAGCCCACCUCGAGAU
GCUACGUGGACGAGGGCAUGCCCAAGACACACCUUAACCCUAGCGGGGGUCGCUAGGGUGAAAUCACACCACGUGAUGGGAGUACGACCUGAUAGGGCGCUGCGGAGGCC
CACUAUUAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUUCAGUGCCGAGGCCACGCGGAGUACGAUCGAGGGUACAGCACUAGGGUCUCUCUUGUUAGAC
CAGAUUUGAGCCUGAGAGCUCUCUGGCUAGCUAGAGAACCCACCGUGGUCUUCACGAAAGAAGAUGGUGAGUUGCUGUUGUGGGAUCCUGGGAAACAGGUUCGGUGAACA
AAAGCCGGUAAUCUCGGGCGUUUGUCAGACGCCGCCGAAUCAACACAGUAUGCAUAUGGGCGGCGCAUGAGAGAAGCCCAGACCAAUUACCUACCCAAAAUGGAGAAAGU
UCACGUUGACAUCGAGGAAGACAGCCCAUUCCUCAGAGCUUUGCAGCGGAGCUUCCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACUGAUAAUGACCAUGCUAAUGCCA
GAGCGUUUUCGCAUCUGGCUUCAAAACUGAUCGAAACGGAGGUGGACCCAUCCGACACGAUCCUUGACAUUGGAAGUGCGCCCGCCCGCAGAAUUGAUGUACCCCAAUAA
UUUGGGAUGCUAAAGUCAUUUAAUGCUGACCUCCACUGGGUGGAUUAAGGUCAAGGUAUGAAGUCCUAUUCGCUCCUGAUAGGAACGACUUCAUAUUGCUUAUAUAUGUG
CUGGUGCACACAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGCCCCUAAGGGAUGCGGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCCUGAGGAAGAGGUGGAC
GGCCAGAUUAGCGCUAACGCAAUUAGUGGAAAAAGAAUUACAGUUAGGACAUAUAGAACCUUCACUUAGUUGUUGGAACACACCUGUCUUCGUGAUCCGGAAGGCUUCCG
GGUCUUACCGCUUAUUGCAUGAUCUGCGCGCUGUUAAUGCCAAGCUUGUUCCCUUUGGGGCCGUCCAACAGGGGGCGCCAGUUCUCUCUGCGCUCCCGCAUGGCUGGCCC
CUAAUGGUCCUAGACCUUAAGGAUUGCUUCUUCUCUAUUCCUCUUGCGGAACAAGAUCGCGAAGCUUUUGCAUUCACCCAGGUCGCCAGUGCGACACCCGUUCCCACACA

ACGGGCACUGGGGUGCAACUCCCCCGUCUAUCCUGGACGUCACCAGGACCAUCCAGUCGAGACCUGAAGUGGGUUUCCUGACGAGGCUGUGGAGAGAGCUUUCGCUUUAC
UCCCGCACAAGCCGAAACUGGAUAGGUGGCUCCAUCUUAGCCCUAGUCACGGCUAGCUGUGAAAGGUCCGUGAGCCGCUUGACUGCAGAGAGUGCUGAUACUGGCCUCUC
UGCAGAUCAAGUGGAAUUUAAUUCGCUUCGAUUCAAGAAUGAGAUACGAAGCAAGAAUAGAAAGCGCUUAUGUGACCGAUGAGUAAUUACAUACAGUUCAAGUCCAAAUG
CCGUAUUGAGCCUGUAUGUCUGCUCCUGAAGCAGACUUCCCCAACUGACACAAACCGUGCAACUUGAAACCCCGCCUGGUCUUUCCAGGUCUAGAGGGGUGACACUUUGU
ACUGUGUUUGGCUCCACGUUCGAUCCACUGGCGAGUGUUAGUAACACCACUGCUGCUUCGUAGCGGAGCAUGACGGCCGUGGGAACUCCUCCUUGGUAACAAGGACCCAC
GGGGCCAAAAGCCACGUCCGAAUGGACCCGUCAUGUGUGCAACCCCGGCACGGUAGCUUUGUUGUGAAACUCACUUUAAAGUGACAUUGAUACUGGUACCCAAACACUGG
UGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGGAACAUGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUGUAAAAGCGCCCAGUUUAAAAAGCUUCUAUGC
CUGAAUAGGUGACCGGAGGCCGGCACCUUUUCUCUAACUACACUUACUCUCCCACGGCCCAAGCCUCGUCUAGGAUGCAAUAGACGAGGUGUAAGGACUAGAGGUUAGAGG
AGACCCCGUGGAAACAUGUUGGGUUCUAGAUGUUAUGAUGACGAGUCGGUCGGCUCCGCGCUAGGUUUGGUCGCCUAGGGGAUGGAGAUAUGGAAAGGGUCUCGUGUGCU
GUCAGUCGGUCGAAAGACGCGCUUGCAACAUGGGCCUAUAACCGGAUAAGUCAUAGCAAUACUAGCCAACAAUAGGUUUAUACCUUCCCAGGUAACAAACCUUAACUUUC
GAUCUCUUGUAGAUCUGUUCUCUAAACGAACUUUAAAAUCUGUGUGGCUGUCACUCGGCUGCAUGCUUAGUGCACUCACGCAGUAUAAUUAAUAACUAAUUACUGUCGUU
GACAGGACACGAGUAACUCGUCUAUCUUCUGCAGGCUGCUUACGGUUUCGUCCGUGUUGCAGCCGAUCAUCAGCACAUCUAGGUUUCGUCCGGGUGUGACCGAAAGGUAA
GAUGGAGAGCCUUGUCCCUGGUUUCAACGAGAAAACAGGGUUCUUUAGUCUGUUUCUCUAUAAGAACACUCAAUUUUCACGCUUUCUGUCUUCUUUCUUCCAGGGCUCUC
CCCUAGCCCUAGGCUCUGGCCGUUGCGCCCGGCGGGGUCAACUCCAUGAUUAGCAUGGAGCUGUAGGAGUCUAAAUUGGGGACGCAAAUGUUUGAGACGUCGCCUUGCAG
UGUUAACUUGGCUCUCAUGAACCUCUCUGAUCUUCCACAAGGGGUAGGCUACGGGUGAAACCUCUUAGGCUAAUACUUCUAUGAAGAGAUGCUUUGGAUAGGGUAACAGC
GGCGGAUAUUGGUGAGUUGUUAAGACAAAAACCAUUCAACGCCGGAGGACUGGCUCUCAUCCAGUGGAUGCAUUGAGUGGAUUGAUUGUCAGGGCUGUCUCUAGGUUUAA
UCUCAGACCUCUCUGUGCUUAGGGCAAACACCAUUUGGCCUUAAAUGGGAUCCUGUGAGAGGGGGUCCCUCCAUUUACAGCUGGACUGUACUUUGGGGCCUUAUGUGGUG
UUUGCCGCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCAUUCUUAAACAAUAAUGUCAUGAUGACCACACAAGGCAGAUGGGCUAUGUAAACGUUUUCGCAAUUCCGU
UUACGAUACAUAGUCUACUCUUGUGCAGAAUGAAUUCUCGUAACUAAACAGCACAAGUAGGUUUAGUUAACUUUAAUCUCACAUAGCAAUCUUUAAUCAAUGUGUAACAU
UAGGGAGGACUUGAGAGAGCCACCACAUUUUCACCGAGGCCACGCGGAGUACGAUCGAGGGUACAGUGAAUAAUGCUAGGGAGUGCUGCCUAUAUGGAAGAGCCCUAAUG
UGUAAAAUUAAUUUUAGUAGUGCUAUCCCCAUGUGAUUUUAAUAGCUUCUUAGGAGAAUGACAAGUGGUUGGACUCCUUCUAAAGUAGAGACAAUUUGAUCUAAUUUGAA
UUGGCUUAACCCUACUGCACUAACCGAACUAGAUAACGGUGCAGUAGGGGUAAAUUCUCCGCAUUCGGUGCGGCACAGGACGCUAGUUGUGAACAAGGUGCGAAGAGCCU
AUUGAGCUACAGAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCCAACCACGGAGCAGGUAGUUGCAAGCCAGCAAUCGACCUGUCGUAACGCGUAAGUCUGUGGCGGAA
CCGACUACUUUGGGUGUCCGUGUUUCCUUUUAAUUUUAACUGGCUGCUUAUGGUGACAAUCAUUGAUUGUUAUCAUAAAGCGAAUUGGAUUGGCCAUCCGGUGUUCACGC
AGAAAGCGUCUAGCCAUGGCGUUAGUACGAGUGUCGUGCAGCCUCCAGGCCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCGCUG
GGGUGACCGGGUCCUUUCUUGGAGCAUCCCGCUCAAUACCCAAAAUUUGGGCGUGCCCCCGCAAGAUCACUAGCCGAGUAGUGUUGGGUCGCGAAAGGCCUUGUGGUACU
GCCUGAUAGGGUGCUCGAGAGUUUUAAAAAAUUUAUUCGCCCUUGCGUGUCCAUUGUUGCAAGGAGCGAUUUGGAGAAAAUAAAUGUUCAUGUCCUACUGUUCAAGCCUC
UAAGCUGUGCCUUAGGUGGCUUUAGGGCAUGGACAGGCAGAUAGGCAUGUAGCUUGAUUACCUACAUGUCUAUCGCCCUGGGAAAGACCAGAGAUCCUGCUGUCUCCCUCA
GCAUCAUUCCAGGCACAGAACGCCAGAAAAUGGAAUGGUGCUGUUGAAUCAACAGGUUCU
>0.05-1
GUCUGUACUAGUUAGCUAACAUGCUCUGUAUCUGGCGGACCCGUGGUGGAACUGACGAGUUCUGAACACCCGGCCGCAACCCUGGGAGACGUCCCAGGGACACCAGGGCA
AAGAGAAGAGUGGUGCAGAGAGAAAAAAGAGCAGUGGGGAUGGGAGCUAUGUUCCUUGGGUUCUUGGGAGCAGCAGGAAGCACUAUGGGCGCAGCGUCAGUGACGCUGAC
GGUACAGGCCAGACAAUUAUUGUCUGGUAUAGUGCAACAGCAGAGCAAUUUGCUGAGGGCUAUUGAGGCGCAACAGCAUCUGUUGCAACUCACAGUCUGGGGCAUCAAGC
AGCUCCAGGCAAGAAUCCUGGCUGUGGAAAGAUACCUAAAGGAUCAACAGCUCCUGGGAAUUUGGGGCUGCUCUGGAAAACUCAUUUGCACCACUGCUGUGCCUUGGAAU
GGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACAUUCCGAGGGGACCGUCCCCUCGGUAAUGGCGAAUGGGACGCAAUGAAUAACCAACGGAAAAAGGC
GAGAAAUACGCCUUUCAAUAUGAAAACAGCUCUGGGGUUGUUCCCACCCCAGACCACGUGGCGGCAGUACUCUGGUAUACGGUACCCUUGUACGCCUGUUUUAUAUAGUA
GGACUAGCAAACGGAGGGACUAGCCGUAGUGGCGAGCUCCCUGGGUGGUCUAAGUCCUGAGUACAGGACAGUCGUCAAUAGUUCGACGUGAGCAGGAGCCCACCUCGAGA
UGCUAAGUGGACGAGGGCAUGCCCAAGACACACCUUAACCCUGGCAGGGGUCGCCAGGGUGAAAUCACACCAUGUGAUGGGGGUACGACCUGAUAGGGUGCUGCAGAGGC
CCACUAACAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUUCAGUGCCAGGGCCACGCGGAGUACGAUCGAGGGUACAGCACUAGGGUCUCUCUUGUUAGA
CCAGAUUAGAGCCUGGGAGCUCUCUGGCUAGCAAGGAACCCACCGUGGUCUUCACGAUAGAAGAUGGUGAGUUGCUGUCGUGGGAUCCUGGGAAACAGGUACGGUGAACA
AAAAGCCGGUAAUCUCGGGCGUUCGUCAGACGCCGUCGUAUCAACACAGUAUGCAUAUGGGCGGCGCAAGAGAGAAGCCCAAACCAAUUACCUACCCAAAAUGGAGAAAG
UUCACGUUGACAUCGAGGAAGACAGCCCAUUCCUCAGAGCUUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACUGAUAAUGACCAUGCUAAUGCC
AGAGCGUUUUCGCAUCUGGCUUCAAAACUGAUCGAAACGGAGGUGGACCCAUCCGACACGAUCCUUGACAUUGGAAGUGCGCCCGCCCGCAGAAUUGAUGUACCCCAUUA
AUUUGGGAUGCCAAAGUCAUUUGAUGCUGACCUCCACUGGGUGGAUUAAGGUCAAGGUAUGAAGCCCUAUUCGCUCCUGAUAGGAUCGACUUCAUAUUGCUUAUAUAUGU
GCUAACGCACAUAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGCCCCUAAGGGAUGCGGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGA
CGGCCAGAUUAGCGCUAACGCAAUUAGUGGAAAAAGAAUUGCAGCUAGGACAUAUAGAACCUUCACUUAGUUGUUGGAACACACCUGUCUUUGUGAUCCGGAAGGCUUCC
GGGUCUUAUCGCUUAUUGCAUGACUUGCGCGCUGUUAACGCUAAGCUUGUUCCUUUUGGGGCCGUCCAACAGGGGGCGCCAGUUCUCUCCGCGCUCCCGCGUGGCUGGCC
CCUGAUGGUCCUAGACCUCAAGGAUUGCUUCUUCUCUAUUCCUCUUGCGGAACAAGAUCGCGAAGCUUUUGCAUUUACUCAGGUCGCCAGUGCGACACCCGUUCCCACAC
AACGGGCAUUGGGGUGCAACUCCCCCGUCUAUCCUGAACGUCAUCAGGACCAUCCAGUCGAGACCUGAAGUGGGUUUCCUGACGAGGCUGUGGAGAGAGCAAAUUGCUUU
ACUCCCGCACAAGCCGAAACUGGAUUGGUGGCUCCAUCUUAGCCCUAGUCACGGCUAGCUGUGAAAGGUCGGUGAGCCGCAUGACCGCAGAGAGUGCUGAUACUGGCCUC
UCUGCAGAUCAUGUGGAAUUUAAUUCGCUUUCGGUUCAAAAAUGAUGAUGACGAAGCAAGAAUAGAUAGCGCUUAUGUGACCGAUGAGCAAUUACAUACAGUUCAAGUCC
AAAUGCCGUAUUGAGCCUGUAUGUCUACUCCUAAAGCAGGUUUCCCCAACUGACACAAACCGUGCAACUUGAAACUCCGCCUGGUCUUUCCAGGUCUAGAGGGGUAACAC
UUUGUACUGUGUUUGGCUCCACGUUCGAUCCACUGGCGAGUGUUAGUAACAACACUGCUACUUCGUAGCGGAGCAUGGCGGCCGUGGGAACUCCUCCUUGGUAACAAGGA
CCCACGGGGCCGAAAGCCACGUCCACAUGGACCCGUCAUGUGUGCAACCCCAGCACGGUAGCUUUGUUGUGAAACCCACUUUAAAGUGACACUGAUACUGGUACCCAAAU
ACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACAAGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUAUAAAAGCGCCCAGUUUAAAAAGCUUC
UAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCUUUAAUUACACUGAAAUUCACGGCCCAAGCCUCGUCUAGGAUGCAAUAGACGAGGUAGAGGGACUAGAGGUUA
GAGGAGACCCUGUUGGGUUCUAGAUGUUAUGAUGACGAGUCGGUUCGGGCUCCGCACUAGGUUUGGUCACCUAGGGGAUGGAGAUAUGGAAAGGGUCUCGUGUGCUGUUA
GACGGUCGUAAGACGCGCUUGCAACAUGGGCCUAUGACCGGAUAAGUCAUAGCAAUACUAGCCAACAAUAUUAGGUUUUUACCUUCCCAGGUAACAAACCACAACUCUCG
AUCUCUUGUAGAUCUGUUCUCUAAACGAACUUUAAAAUCUGUGUGACUGUCACUUAGCUGCAUGCUUAGUGCACUCACGCAGUUUAAUUAUAAUUAAUUACUGUCGUUGA
CAGGACACGAGUAACUCGUCUAUCUUCUGCAGGUUGCUUACGGUUUCGUCCGUGUUGCAGCCGAUCAUCAGCAUACCUUGGUUUCGUCCGGGUGUGACCGAGAGGUAAGA
UGGAGAGCCUUGUCCCUGGUUUCAACGAGAAAACAGGGUUCUUCAAUCUGUUUCUCUAUAAGAACACUCAAAUUUCACGCUUUCUGUCUUCUUUCUUCCAUGGCUCUCCC
CUUGCCCUAGGCACAGGGCGUCGCGCCCGGCGGGGUCAACUCCAUCAUUAGCAUGGAGCUGUAGGAGUCUAAAUUGGGGACGCAGAUGUUUGGGACGUCACCUUGCAGUG

UUAACUUGGCUUUCAUGAACCUCUUUGAUCUUCCACAAGGGGUAGGCUACGGGUGAAACCCCUUAGGCUAAUACUUCUAUGAAGAGAUGCCUUGGAUAUGGUAACAGCGG
CGGAUAUUGGUGAAUUGUUAAGACAAAAACCAUUCAACGCCGAAGGACUGGCUCUCAUCCAGUGGAUGCAUUGAGUGAAUUGAUUGUCAGGGCUGUCUCUAGGUUUAAUC
UCAGACCUCUCUGUGCUUAGGGCAAACACUAUUUGGCCUUAAAUGGGAUCCUGUGAGAGGGGGUCCCUCCAUUGACAGCUGGACUGUUCUUUGGGGCCUUAUGUAGUGUU
UGCCUCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCACUCUUAAACAACAAUGGUGCUGACUCAACUCAGGCCUAAACUCAUGCAGACCACACAAGGCAGAUGGGCUA
UAUAAACGUUUUCGCUUUUCCGUUUACGAUAUAUAGUCUACUCUUGUGCAGAAUGAAUUCUCGUAACUACAUAGCACAAGUAGAUGUAGUUAACUUUAAUCUCACAUAGC
AAUCUUUAAUCAGUGUGUAACAUUAGGGAGGACUUGAAAGAGCCACCACAUUUUCACCGAGUGUACAGUGAACAAUGCUAGGGAGAGCUGCCUAUAUGGAAGAGCCCUAA
UGUGUAAAAUUAAUUUUAGUAGUGCUAUCCCCAUGUGAUUUUAAUAGCUUCUUAGGAGAAUGACAAGUGGUUGGACUCUUUCUAAAUUGGAGACAAUUUGAUCUGAUUUG
AAUUGGCUUAACCCUACUGCACUAACCGAACUAGACAACGGUGCAGUAGGGGUAAAUUCUCCGCGUUCGGUGCGGCACAGGACGCUAGUUGUGAACAAGGUGUGAAGAGC
CUAUUGAGCUACAGAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACCACGGAGCAGGUGGUUGCAACCCAGCAACCGGCCUGUCGUAACGCGCAAGUCUGUGGCGG
AACCGACUACUUUGGGUGUCCGUGUUUCCUUUUUAAAUUUUGUGGCUGCUUAUGGUGACAAUCAUUGAUUGUUAUCAUAAAGCGAUUUGGAUUGGCCAUCCGGUGGCCAU
GGCGUUAGUAUGAGUGUCGUGCAGCCUCCAGGCCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCGCUGGGGUGACCGGGUCCUUU
CUUGGAACAACCCGCUCAAUACCCGAAAUUUGGGCGUGCCCCCGCGAGAUCACUAGCCGAGUAGUGUUGGGUCGCGAAAGGCCUUGUGGUACUGCCUGAUAGGGUGCUUG
CGAGUUUUUAAAUUUUUAUUCGCCCUUGCGUGUCCAUUGUUGCAAGGAGCGAUUUGGAGAAAAUAAAUGUUCAUGUCCUACUGUUCAAGCCUCCAAGCUGUGCCUUGGGU
GGCUUUAGGACAUGGACAGGCAGAUAGGCAUGUAGCUUAGCACCUACAUGUCUAUCGCCCUGGGAGAGACCAGAGAUCCUGCUGUCUCCCUCAGCAUCAUUCCAGGCACAG
AACGCCAGAAAAUCGAAUGGUGCUGUUGAAUC
>0.1-0
UGACUAGCGGAGGCUAGAAGGAGAGAAAAACCAAAAGAAACACCAACCGCCGCCCACAGGACGUCAAGUUCCCGGGCGGUGGUCAGAUCGUCGGUGGAGUGUACUUGCUG
CCGCGCAGGGGCCCCCGGUUGGGUGUGCGCGCAACAAGGAAGACUUCAGAACGAUCCCAGCCCAGAUACAGCGGGGGAGACAUAUAUCACAGCCUGUCUCAUGCCCGACC
CCGCUGGUUAGCUCGCCAGUUAGCGAGGUCUGUCCCCACAGGACAGAUAAUCGGGUGCAACUCCCGCCCCUCUUCCGAGGGUCAUCGGAACCUGUAACGGGGCAAGGCUU
GACCGAGGGGACUAUAACAUGUAUAGGCGUCAAGCGGGGCUUCGGUUGUACGCGGAUAGGAAUCCCCUCAGGGAAAACUCACCGCAACGAAACAAAUCGUUCAGAAACAC
UUGACCACGAGGGGUCCUCUAUAGUACCGUCAGGUUGCGGCAGCCCCCGUGCGAUAGGCUAACUGUGGUGUUCCCCGCACUCCGUCGAGCGGUUAAUACGACGCUUAUUA
AGACAGGGGGAUACGUGUGGUGCCGUCCGCUCCCCCUAUGGAAUAACUGACGCACCACAACCCCGCAAGGAAGGACUAGGCUGUCCUUGGGUGCUAACGACACUCCGGCC
CCAAUUGGCGGCGUAGUACACACUAUUGAAUUAAACAGCCGACCAAUUGCACUACCAUCACAAUGGAGAAGCCAGUAGUUAACGUAGACGUAGACCCUCAGAGUCCGUUU
GUCGUGCAACUGCAAAAGAGCUUCCCGCAAUUUGAGGUAGUAGCACAGCAGGUCACUCCAAAUGACCAUGCUAAUGCCAGAGCAUUUUCGCAUCUGGCCAGUAAACUAAU
CGAGCUGGAGGUUCCUACCACAGCGACGAUUUUGGACAUAGGCAGCGCACCGGCUCGUAGAAUCAGGUCUCUUGAAUCUGUUCCACUUUAUGGACACUCAUUUCACGCUU
UCUGUCUGCUUUUCUUCCAGGGCUCUCCCCCUUGCCCUAGGCUCUGGCCGCUGCGCCCGGCGGGGUCAACUCCAUGGUUAGCAUGGAGCUGUAGGAGUCUAAAUUGGGGAC
ACAGAUGUUGGGAACGUCGCCUUGCAGUGUUAACUUGGCUUUCAUGAAGCUUUUUGAUCUUCCACAAGAGGUAGGCUACGGGUGAAACCUCUUAAGCUAAUACUUCUAUG
AAGAGAUGCUUUGGAUAGGGUAACAGCGGCGGAUAUUGGUGAGUUGUUUGACAAAAACCAUUCAACGCCGGAGACUGGCUCUCAUCCAGUGGAUGCAUUAAGUGGAUUGU
CUGUCAGGGCUGUCUCUAGGUUUAAUUCCUGGCCUCUCUGUGCUUAGGGUAAACAAAACUUGGCCUUAAAUGAGGUCCUGUGAGAGGGGACCUCACCAUUGAAUGCUGGA
CAUUUCUUUGGGGCCUUAUGUUUUGUUUGCCUCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCAUCAAUAAAUAAUUAUGCCAAUGUCAGACCACGCUACGGCGUGCC
ACUCUGCGGAGAGUGCAGUCUGCGAUAGUGCCCCAGGAGGACUGGGUCCACGGCCCAAGUCUCGUCCAGGAUGCAAUGGACGAGAUGUAAGGACUAGAGGUUAGAGGAGA
CCCCGUGGAAACAGCUGUGGGCCGGCUUUUUUACUUGUCUGCGGCGCCUUGGGCGCCGCAGCGUACGAACACACAGCAGUGAUGCCGGCUCGCGCCCGUAGUCUGGAGAA
GCAUCGCCAGGGUUGAGUCGCGGCAGAACCCGGUUCGCGGACGGCCGCGGCGAGCGGGAAUUGGUCACCCCGCCUAUUUAAAGACCCACAGCCAGCCGACUUCUCCAGUU
ACGGGAGCGAGCCCCCAUGGGACGCUUCAAUACUGACAUGGUGUGAAGAGUCUAUUGAGCUAAUUGUAGUUCGGCCCCUGAAUGCGGCUAAUCCUAACCGCGGAGCAGAU
ACCCACAUGCCAGUGGGAAGUCUGUCGUAAUGGGCAACUCUGCGGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCCUUUAUUUUGUACUGGCUGCUUAUGGUGACAAU
UGAAAGAUUGUUACCAUAUAGCAUUGGAUUGGCCAUCCAGUGUUAGGGAGGUCUGGCCUUCCACAGGGGGGCCAGGGAACUUCUUGGUUUGUGAGAAGGUUGGAGUUGCC
CACCGGUUGGGGUAUGACGGUUUCCUGUCAUACUACAGCGGCGCGAAGCUCCGUACUUACACACGAGCUGCCUCCAGGCCUUAAAUGACCUGGUCUCCAAGGCCCUGGAG
GCAUAAGUCAGUGUUAAUAGGACCUACCCGUCACAUAAUUGGAGAAUUGAUGACUCGAUUGGUGCACUUAAUUUUCCAUUGUCCUAUGAAACUGACCAGUUGGACCAUGG
GGAAAGAAGCCCCGCAACUUCCCCGUGGUCCCAAGCGGCCACCCGCACCACAUGCGGAAGAACGACUGCUGGAACGGGGAGUCCUGAAACAGUCAAACCUGUCCUUGAUC
UUACUAGCUCUAUUAUAUACCGGGUGGAAUGAAUAACAUGUUCUUUUGCUCUGUGCAUGGAGAUGCCACCAUGCGCUCUCGGGCUCUUCUGUUUUUGCUCUUCGUGCUUC
UGCCUAUGCUGCCCGCCGUGGUCUUCGCGAUAGAAGAACGCAGGUCUCCCAACUGACACAAACCGUGCAGUUUGAAACUCCGCCUGGUCUUUCCAGGUCUAGAGGGGUGA
CAAUUUGUACUGUGCUUGACUCCACGCUCGGUCCACUGGCGAGUAUCAGUAACAGCACUGAUGCUUCGUAGCGGAGCAUGGCGGCCGUGGGAACCCCUCCUUGGUAACAA
GGACCCACGGGGCCGAAAGCCACGUCCUAACGGACCCGUCAUGUGUGCAACCCCGGCACGGCAGCUUUUCUGUGAAACUCACUUUAACGUGACACUGAUACUGGUACUCA
AACACUGGAGACAGGCUAAGGAUGUCCUUCAGGUACCCCGGGGUAACACGCGACACCCGAGAUCUGAGAAGGGGACUGGGGCUUCUAUAAAAGCGCCCAGUUUAAAAAGC
UUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUCCUUUACAACCACUGAUUUAUGAACAACCAACGGAAAAAGACGGGUCGACCGUCUUUCAAUAUGCCUCCCGG
GAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUUGCCAGGACGACCGGGUCCUUUCUUGGAUCAACCCGCUCAAUGCCUGGAGAUUUGGGCGUGCCCCCGC
AAGACUGCUAGCCGAGUAGUGUGGGGUCGCGAAAGGCCUGGUGGUACUACCUGAUAGGGUGCUGGCGAGUGCCUAAAACAGCUGUGGGUUGCACCCACCCACAGGGCCCA
CUGGGCGCUAGCACUCUGGUACUGAGGUACCUUUGUGCGCCUGUUUUUACGGUCUCUCUGUUAGACCAGAUCUGAGCCUGAGCUCUCUGCUAACUAGGAACCACCUGGGA
GAGACCAGAGAUCCUGCUGUCUCUACAGCAUCAUUCCAGGCACAGAACGCCAGAAAAUCGAAUGGUGCUGUUGAAUCCAGUGCCGGGGCCACGCGGCGUACGAUCGUGGG
UACAGCACUACGUGGUGCAUACGAUAAUGCAACUGAGCAAUUACAUACAGUUCAAGUCCAAAUGCCGUAUUGAGCCCGUAUGUUUGCUCCUACUUGGUAACUAGAGAUCC
CUCAGACUAUUUUAGUCAGUGUGAAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACUAGAAAGCGAAAGUAAACCAGAGAGUCCCUAACGAUUGCGAACACCUGAAUGAA
GCAGAAGGCUUCAUUUGGUGACCCCGACGUGAUCGUUAGGGAAUUUUAAACUGUUAGCCGCCAGCGGCUUGACCCGCUGUGGCCGCGGCGGCCUAGUUGUGACUGAAACG
GCGGUAUGUACAUGUCCCACUUUUCAAGCCUCCAAGCUGUGCCUUGGAUGGCUUUGGGGCAUGGACAAUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGC
AACAUUCCGAAGGGACCGUCCCUUCGGUAAUGGCGAAUGGGACCCAAAAUGGUUGGACUCUUUUUAGAUUAGAGACAAUUUGAAUAAUUUAAAUUGGCUUAACCCUACUG
CAUUAACCGAACUAGACAACAAUGCAGUAGGGGUAAAUUUUCCGCGUUCGGUGCGGAAAAAAAAAAGGAAGAUAGGCAUGUAGUUUGAUUACCUACAUGUCUAUCGCCGA
GUUAGACACGGUUGAUCUCACCCUUCGGGGGGGCUAUAGAGAUCACUGGAAGCACUACCGGACAACCGGAACAUUGCAGCAAUGCAGCCCAGUGUUUUUCCCUCCACUUA
AAUCGAAGGGUAUGGGAAACUCCCCCUCCUAUAACCCCCCCGCUGGCAUCUCCCCCUCAGACUGGCAAAGAGUCUGUGCUAAGCACACUGACGAGUCUCUGAAAUGAGAC
GAAACUCUUUCACUCAUCGCCCAAGGAGGGAAGGCAGCCAUUUGUGGCCGCUACCUCUUCAACUGGGCGGUGAAGUACUUGCGCGGCGCUAUAGCGCCGCGCUAGUGGGC
CCCCGCGCGAAACCCGCAUCAGCCCACUAGAUUGGCCAGGAGGAAAGAGAAAAUAUAGGCUGAAACAUCUAGUAUGGGCAAGCAGGGAGCUGGAAAGAUUCGCACUUGAC
CCUGGCCGGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGACGGCCAGAUAUAUUAGGUUUUUACCUACCCAGGAAAAGCCAACCAAUCUC
GAUCUCUUGUAGAUCUGUUCUCUAAACGAACUUUAAAAUCUGUGUAGCUGUCGCUCGGCUGCAUGCCUAGUGCACUUACGCAGUAUAAAAUAUUAAUAACUUUACUGUCGC
UGACUGGAUACGAGUAACUCGUCCUUCUUCUGCAGACUGCUUACGGUUUCGUCCGUGUUGCAGUCGAUCAUCAGCAUACCUAGGUUUCGUCCGGGUGUGACCGAAAGGUA

AGAUGGAGAGCCUUGCUCUUGGUGUCAGCGAGAAAA

>0.1-1

UGACUAGCAGAGGCUAGAAGGAAAAAAGAAACCAAAGAAACACAAACCGUCGCCCACAGGACGUCAAGUUCCCGGGUGGCGGUCAGAUCGUUGGUGGAGUCUACUUGCUG
CCGCGCAGGGGCCCCCGGUUGGGUGUGCGCGCGACGAGGAAGACUUCUGAACGGUCCCAGCCCAGAUACAGCGGGGGAGACAUUUAUCACAGCGUGUCUCAUGUCCGGCC
CCGUUGGUCAGCUCGCCAGUUAGCGAGGCCUGUUCCCACACAACAGGUAAUCGGGUGCAACUCCCGCCCCUACUCCGAGGGUCAUCGGAACCAUGUAACGGGGCAAGGCU
UGACUGAGGGGACUGCAGCAUGUAUAGGCGCUGGGCGGGGCUUCGGUUGUACGCGGAUAGGAAUCCCCUCAGGGAAAACUCACCGCAACGAAACGAAACAAAUCGUUCAG
AAAUACUUGACCACGAGGGCCCCCCUAUAGUCCAGUUUUGGUGCGGCGGCCCCGUGCGAUAGGCUAACUGCGGUAUUCCCCGCACUCCGUCGAGCGGUUAAUACGACGCU
UACCAAGACCGGGGGAUACGUGUGGUGCCGUCCGCUCCCCCUAUGGCAUAACAAAUGCACCACAACCCCGUAAGGAAGGACAAGGCUGUCCUUGAGUGCCAACGACGCUC
CGGCCCCAAUUGGCGGCGUAGUACACACUAUAAAGAAUCAAGCCGACCAAUUGCACUACCAUCAUCAUGGAGAAACCUGUAGUUAACGUUGACGUAGACCCUCAGAGUCC
GUUCAUUACGCAACUGCAGAAGAGUUUCCCGCAGUUUGAGGUAGUAGCACAGCAGGCCACACCAAAUGACCAUGCAAAUGCCAGAGCUUUUUCGCAUCUGGCCAGCAAAC
UGAUCGAGCUGGAGGUCCCGACCACAGCGACGAUCUUGGACAUCGGCAGCGCACCUGCUCGUAGAAUCAGGGUUCUUUAGUCUGUUUCUCUAUAAGAACACUCAAUUUUC
ACGCUUUCUGUCUUCUUUCUUCCAGGGCUCUCCCCUUGCCCUAGGCUCUGGCCGUUGCGCCCGGCGGGGUCAACUCCAUGAUUAGCAUGGAGCUGUAGGAGUCUAAAUUG
GGGACGCAGAUGUUUGAGACGUCGCCUUGCAGUGUUAACUUGGCUCUCAUGAACCUCUUUGAUCUUCCACAAGGGGUAGGCUACGGGUGAAACCUCUUAGGCUAAUACUU
CUAUGAAGAGAUGCUUUGGAUAGGGUAACAGCGGCGGAUAUUGGUGAGUUGUUAAGACAAAAACCAUUCAACGCCGGAGGACUGGCUCUCAUCCAGUGGAUGCAUUGAGU
GGAUUGAUUGUCAGGGCUGUCUCUAGGUUUAAUCUCAGACCUCUCUGUGCUUAGGGCAAACACCAUUUGGCCUUAAAUGGGAUCCUGUGAGAGGGGGUCCCUCCAUUUAC
AGCUGGACUGUACUUUGGGGCCUUAUGUGGUGUUUGCCGCUGAGGUACUCAGGGGCAUUUAGGUUUUUCCUCAUUCUUAAACAAUAAUGCCAAUGUCAGACCACAUUUCG
AUGUGCCACUCUGCGGAGAGUGCUGUCUGCGAUAGUGCCCCAGGUGGACUGGGUCCACGGCCCAAGCCUCGUCUAAGAUGCAAUAGAUUAGGUGUAAGGACUAGAGGUUA
GAGGAGACCCCGUGGAAACAGCUGCGGUCCGGCUUUUUUACUUGUCUGCGGCGCCUUGGGCGCCGCAGCAUACGAGCACACGGCUGUGAUGUCGGCUCGCGCCCGUAGUC
UGGACAAGCAUCACUAGGGUUAAGUUACAGCAGAACCCGGUUCGCGGACGGCCGCGGCAAGCGGGACUUAGUUACCCCGCCAAUUUAAAGACCCGCAGCCAGCCGACUUU
UCCAGUUACGGGAGCGAGCCCCCAUGAGACGCUUCAAUACUGACAUGGUGCGAAGAGUCUAUUGAGCUAACUGUAGUUCGGCCCCUGAAUGCGGCUAAUCCUAACUGCGG
AGCAGGCGCUCGCAAACCAGUGAGUAGCCUGUCGUAACGGGUAACUCUGCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCUUUUAUCUUAUAUUGGCUGCUUAUGG
UGACAAUUGAGAGAUUGUUACCAUAUAGCAUUGGAUUGGCCAUCCGGUGUUUUUUAGGGAAAUUUGGCCUUCCACAGGGGGGCCAGGGAAUUUCUUAGUUUGUGAGAAGG
UUGGAGUUGCCCACCGACUUGGGUAUGAUGGAUUUCUGUCAUACUACAACGGUGCGAAGCUCCGUACUUACACACGAGCUGCCUCCAGGCCCUAAUAGACCUGGUCUCCA
AGGCACUGGAGGCAUAGGACAGUAUUAGUAGGACCUACCCGUCACAUAAUUGGCGAAUUGUUGACCCGAUUGGUGUACUUAAUUUUCCAUCGUCCUAUGAAACUGACCAG
AUGGGUCCAUGGGGAAAGAAGCCCCGCAACUUCCCCGUGGCCCAAAGCGGCCAUCCUCACCACAUGCGGAAGAACGAGUGCUGGAACGGGGAGACUUGAGACAAUUAAGC
CUGUUCUUGACCUUACUAGCUCCAUCAUAUAUCGGGUGGAAUGAAUAACAUGUUCUUUUGCUCUGCGCAUGGAGAUGCCACCAUGCGCUCUCGGGCUCUUCUGUUUCUGC
UCCUCGUGUUUUUGCCUAUGCUGCCCGCCGUGGUCUUCACGAAAGAAGAAAGCAGGUUUCCCCAACUGACACAAACCGUGCAACUUGAAACUCCGCCUGGUCUUUCCAGG
UCUAGAGGGGUAACACUUUGUACUGUGUUUGGCUCCACGUUCGAUCCACUGGCGAGUGUUAGUAACAACACUGCUACUUCGUAGCGGAGCAUGGCGGCCGUGGGAACUCC
UCCUUGGUAACAAGGACCCACGGGGCCGAAAGCCACGUCCACAUGGACCCGUCAUGUGUGCAACCCCAGCACGGUAGCUUUGUUGUGAAACCCACUUUAAAGUGACACUG
AUACUGGUACCCAAAUACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCGAGGUAACAAGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUAUAAAAGCGCC
CAGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCUUUAAUUACACUGAAAUUAUGAACAACCAACGGAAAAAGACGGGAAAACCGUCUAUC
AAUAUGAUGGCGUUAGUACGAGUGUCGAACAGCCUCCAGGCCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCCCCAGGACGACCG
GGUCCUUUCUUGGAUUAACCCGCUCCAUGCCUGGGUAUUUGGGCGUGCCCCCGCGAGACUGCUAGCCGAGUAGUGUUGGGUCGCGAAAGGCCUUGUGGUUUAAAACAGCC
UGUGGGUUGUUCCCACCCACAGGGCCCAUGGGCGCUAGCACACUGGUAUCGUGGUACCUUUGUGCGCCUGUUUUAUACGGUCUCUCUGUUAGACCAGAUAGAGUCUGAAC
UCUCUGCUAGCGAGGAACCACCUGGGAAAGACCAGAGAUCCUGCUGUCUCUGCAACAUCAAUCCAGGCACAGAGCGCCGCAAGAUCGAAUGGUGCUGUUGAAUCCAGUGC
CGAGGCCACUCGGAGUACGAUCGAGGGUACAGCACUAGCUGGUGCAUACGAUAAUGCAGAUGAGUAAUUACAUACAGUUCAAGUCCAAAUGCCGUAUUGAGCCUGUAUGU
CUGCUCCUGCUUGGUAGCUAGAGAUCCCUUAGACCCUUUUAGUCAGUGUGGAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACUUGAAAGUGAAAGGGAACCAGAGAUUC
CCUGACGACUACGAACACCUAAAUGAAGCGGAAGGCUUCAUUUGGUGACCCCGACGUGAUAGUUAGGGAAUUUUAAACUGCUAGCCGCCAGCGACUUGACCCGCUGUGGU
CGCGGCGGCUUGGUUGUUACUGAAACAGCGGUAUGUUCAUGUCCUACUGUUCAAGCCUCCAAGCUGUGCCUUGGGUGGCUUUAGGACAUGGACAGUGGCCGGCAUGGUCC
AGCCUACUCGCUGGCGCGGGCUGGGCAACAUUCCGAGGGGACCCGUCCCUCGGUAAUGGCGAAUGGGACCCAAAGUGGUUGGACUCUUUUUAAAUUAGAGACAAUUUGAA
CUAAUUUUAAUUGGCUUAACCCUACCGCACUAACCGAACUAGAUAACGGUGCAGUGGGGGUAAAUUCUCCGCAUUCGGUGCGGGCAGAUAGGCAUGUAGCUUAGCACCUA
CAUGUCUAUCGCCUUGCCAGACACGGUUGAUCUCACUCUCCGGGGGGGCUAUAGAGAUCGCUGGAAACACUACCGGACAACCGGAACAUUGCAGAAAUGCAGCCCAGUGG
CUAUCCCUCCACUUAAAUCGAAGGGUAUGGGAAAUUCCCCUUCUUAUAACCCUCCCGCUGGUAUCUCCCCCUCAGACUGGCGAUGAGUCUGUGCUAAGCACACUGAUGAG
UCUCUGAAAUGAGACGAAACUCAUGCACUCAUUGCCCAGGGAGGGAGAGCGGCCAUUUGUGGACGCUACCUCUUCAACUGGGCGGUGAAGCACCUGCGCGGCGCCAUAGC
GCCGCGCUAGUGGGCCCCCGCGCGAAACCCGCACUAGCCCACUAGAUCGGCCAGGAGGAAAGAAAAAAUAUAGACUAAAACAUCUAGUAUGGGCAAGCAGGGAGCUAGAA
CGAUUUGCACUUAAUCCUGGCCAGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCUUGAGGAAGAGGUGGACGGCCAGACAUAUUAGGUUUUUACCUUCCCAGGUA
ACAAACCAACUAACUCUCGAUCUCUUGUAGAUCUGUUCUCUAAACGAACUUUAAAAUCUGUGUGACUGUCACUUAGCUGCAUGCUUAGUGCACUCACGCAGUUUAAUUAU
AAUUAAUUACUGUCGUUGACAGGACACGAGUAACUCGUCUAUCUUCUGCAGGUUGCUUACGGUUUCGUCCGUGUUGCAGCCGAUCAUCAGCAUACCUUGGUUUCGUCCGG
GUGUGACCGAGAGGUAAGAUGGAGAGCCUUGUCCCUGGUUUCAACGAGAAAA

>0.15-0

GGGUCUCUCUUAGUAGACCAGAUAGAGCCUGGGAGCUCUCUGGCUAGUAAGGGAACCCAUGGGAAACGGGCAGGCGGCGGCGACCGCCGAAACAACCGCAGGGAAACGGA
CUGAGGGGCCAGCCCCAGGCCCCGAAACAAGCUUAUGGGGCGCUGGGAAAGACCAGAGAUCCUGCUGUCUCUGCAACAUCAAUCCAGGCACAGAGCGCCGCAAGAUCGAA
UGGUGCUGUUGAAUCUACAACGGGGGAGACAUUUAUCACAGCGUGUCUCGUGUCCGGCCCCGUUGGAGUCCCUAACGAUUGCGAACACCUGAAUGAAGCAGAAGGCUUCA
UUUGGUGACCCCGACGUGAUCGUUAGGGAAUAUGGCGUUAGUACGAGUGUCGUGCAGCCUCCAGGCCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAG
UACACCGGAAUCGCCGGGUUGACCGGGUCCUUUCUUGGAACUACCCGCUCAAUGCCCGGAAAUUUGGGCGUGCCCCCGCAAGACUGCUAGCCGAGUAGCGUUGGGUUGCG
AAAGGCCUUGUGGUACUUUGAGCACGGUUGAUCUCACCCUCCGGGGGGGCUAUAGAGAUCGCUGGAAACACUACCGGACAACCGGAACAUUGCAGAAAUGCAGCCCAAGC
UUCUGUCCAGAGGAGGCAGGGCUGCCAUAUGUGGCAAGUACCUCUUCAAUUGGGCAGUAAGACUUUGCCUACUCCUACUCAGCGUAGGGGUAGGCAUCUUGUAACGGCGC
AAGGCUUGACCGAGGGGACUAUAACAUGUAUAGGCGAAAAGCGGGGUCUCGGUUGUAACGCGCUUAGGAAGUCCCCUCGAGGUGCCAAGGUAUCUUUACGUCUACAUUGC
UGUUGUCGGCUUUGACUGUACCUUUGGUAUAAGUUCUCGAUCUUUAAAAUCGUUCCAGGACAUACUGAUGAGGAUGUCAAAAAUGCAGUUGGGGCCCUCGAUAAGUACAG
UAUUAGUGGGACCUACACCUUCAGUAUUGGAAGAAAUCUAAUGUCUCAAUUGGUUGCACUUUAAAUUUUCCAUUAGUCCAUUGAAACUGUACCAGUUCCUAGAGAGCUUG
GCCGUCGGGCCUUAUACCCGACUUGCUGAGUUUCUCUAGGAGAGUCCCUUUCCCAGCCCUGAGGCGGCUGGUCAAUAAAAGCCUCAAAGUAAGUGGACUUUAAGUUUUUC
UGCUAAGCCCAACAUUCCACCACACCCAGGCACACACUACACACACCCGCCCGACUUAGUUAAAACAGCUGUGGGUUGUCACCCACCCACAGGGUCCACUGGGCGCUAGC

ACUCUGGUACCUCGGUACCUUUGUGCGCCUGUUUUAUACGGACUCGGCUUGCUGAGGUGCCACAGCAAGAGGCGAGAGAGUUGUUAGUCUACGUGGACCGACAAGAACAG
UUUCGACUCGGAAGCUUGCUUAACGUAGUGCUGACAGUUUUUUAGGGAAGAUCUGGCCUUCCCACAAGGGGAGGCCAGGGAAUUUUCUCACACCUGACUCCAGACUCUUC
AGUCUGAGUUUGUGGAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUACUGUGUCGUGCCUUUUUUAGUCAUGGCCGGCGCCGCAGCGCCGGCCUACGAGCACGCGACC
ACGAUGCCUUUCCCUGAAGAGACGAAGUGAUCAAGAGAUUGAAGACGAGUGAACUAACGACUCCUCCUUCUCUCACAAGUCGAAACUCAGAGUCGGAAAGGGAAUUUAAU
UCGUUAUCGAUUCAAGAAUGGGUAUAACGAAGUAAGAAUAGAAAGCGCUUAUGUGACCGCCUCCAGGCCCUGACUGACCUGGUUUCCAGGGCCCUGGAGGCGGCCGACAG
GAGUAAUUCCUGCCGAAACAGGGUUUUCACCCUUCCUUUCUUCAGGUGUCCUUCCUCGCGCCCGCAGGACGGAAGCCGCGGCCACGCCGAGUAGGCACGAGGGUACAGCU
UCCAUGGGACGCUUCAAUACUGACAUGGUGUGAAGAGUCUAUUGAGCUAAUUGGAUGUCUCCGGCCCCUGAAUACGGCUAAUCCUAAUCGCGGAGCAGAUACCCACGCAC
CAGUGGGCAGUCUGUCGUAACGGGCAACUCUGCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCUUUUAUUCUUGUACUGGCUGCUUAUGGUGACAAUUGAGAGAUU
GUUACCAUAUAGCUAUUGGAUUGGCCAUCCAGUGUUUGGCCCAUGGGGAAAGAAGCCUCGCAACUUCCCCAUGACCCAAGGCUCGACUCCGUGGCCUGGGGGAACGUGAA
CGGGUUGGGUCGCGGUGUACCCCGGUUCGAGUCCAAAGCUAAGCAAUCACACUCGGAUCGGCCGGAGCCGCGGCUAACGUGGUAUUGGCUAUCCCGUCUGACCCAGCGAC
GAAUAUCCAGGGUACGGAGUAGAGUCGUUAUGGGCGGCGCAUGAGAGAUUAGCCCAAUACCAAAUCAUACUACCCAAAUAUGGAGAAAGUUCACGUUGACAUCGAGGAGG
AUAGCCCCUUCCUCAGAGCUUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACCGAUAACGACCAUGCUAAUGCUAGAGCCUUUUCGCAUCUAGCU
UCUAAGUUGAUCGAAACGGAGGUAGACCCAUCCGAUACGAUCCUAGACAUAGGCAGUGCACCUGCCCGUAGAAUAGAAAAACCAAACGUAACACCAACCGCCGCCCAAUG
GACGUUAAGUUCCCGGGUGGCAGCCAGAUCGUUGGCGGAGUUUACUUGUUGCCGCGUAGGGGCCCUAGAUUGGGUGUGCGCGCGACUAGGAAGACUUCGGAGCGGUCGCA
ACCUCGUUCGGGUCGGCAUGGGAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUCCGUAAGGAGGACGUAGUCCCCUCGGAUGGCAAAGGGAGAUCCACCGGUCUCCGAG
GAGGUCGGGGUUGCCCACCGUUUGGGUAUGAUGGAUCUAUCAUACUACACGGUGCGAAGCUCCGUACUUACACGCGGGCUCACGGCCCAGCCUGACUGAAGCUGUAGGUC
AGGGGAAGGACUAGAGGUUAGUGGAGACCCCGUGCCACAAAGCGGCCACCCGCACCACAUGCGGAAGAACGACUGCUGGAACGGGGAGUCCUCCUGACCCUGGGGGAGGC
CAUGGUUCAGGGGCCAAGGGGAAAGAAAAGAUAUAAGUUAAAACAUAUAGUAUGGGCAAGCAGGGAGCUAGAACGAUUUGCUGUCAAUCCUAGCCUGACUAGCGGAGGCU
AGAAGGAGAAGUGGUUGGACUCUUUUUAAAUUAGAGACAAUUUAAACUAAUUUGAAUUGGCUUAACCCUACUGCACUAACCGAACUAGAUAACGGCGCAGUAGGGGUAAA
UUCUCCGCAUUCGGUGCGGCCCCAGUCCGACAGGUCAUGGCAUGACCCCAUGGUGCUGAGAGGGCAGAAAACUCACCGAAACGAAACGAAAACAAUCGUUCAGAAAUACU
UGACCACGAGGGGCCCCCUACUAUCUGCGUUGGCGGUGCGGCAGCCCCCGUGCGACAGGCUAACCGCGGUAUUCCCCGCACUCCGUCGAGCGGUAAAUACGACGCUUACC
AAGACUAAACGUUUCACCGGUUUGUAAAUUGGAACUUCGUAGUUCGCCAGACGCCUUCUUCGUAUCAAGGAACUGGAAAUUGUACGGUGAGUCUGUCAUUUUAUUUUCUA
CUUUUCGUACCAGUCUUUAAAAAAUUUAUUCGCCCUUGCGUGUCCAUUGUUGCAAGGAGCGAUUUGGAGAAAAUAAACUUUGGUAGCUAGAGAACCUUAGACCCUUUAAG
UCAGUGUGGAAAAUCUCUAGCAGUGGCGCCCGACAGGGACUUGAAAGCGAAAAGAGAACCAGAGCAUCAAUAACUACAUACAGUUCAAGAGCAAACACCGUAUUGAACCA
GUAUGUUUGUUAGUGAGAGUUUUUCCCUCCUCUUAAAUCGAAGGGCAAGCAGGUUUCCACAACUGAUAAAACUCGUGCAUUUUGAAACCCCGCCUGGUCUUUCCAGGUCU
AGAGGGGUGACACUUUGUACUGUGCUCGACUCCACGCUCGGUCCACUGGCGGGUGUUAGUAGCAGCACUGUUGCUUCGUAGCGGAGCAUGGUGGCCGUGGGAACUCCUCC
UUGGUGACAAGGACCCACGGGGCCAAAAGCCACGUCCAGAUGGACCCACCAUGUGUGCAACCCCAGCACGGCAACCUUACUGCGAACACCACCUUAACGUGACACUGAUA
CUGGUACUCGGUCACUGGUGACAGACUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGGGACACUCGGGAUCUGAGAAGGGGAUUGGGACUUCUUUAAAAGUGUCCAA
UUUAAAAAGCUUCUAUGUCUGAAUAGGCGACCGGAGGCCGGCGCCUUUCCAUUACCUACUACUAAAUCUGAGCUGAUCGUUGUUGUUAUUCUUCAAUUUCUGCAGGUAAC
UUAGGAAGCGUUUCUUGAGCUUCCCUGGGAUGAGCGUUUGGGAGAGCUGAUUCUGCAGCCCAGAGAGUAGUCUCAGGGCAUCCUCUGGAGCCUGACCUGUGACGUCGCAU
CAUAGACCGCCAGUAGACCUGGGAGCAGAUUCACCGCCGCGGCCGUCUCCUUUAAGGUGCUGUGGUCUGUACUAGUUAGCUAACUAGAUCUGUAUCUGGCGGUUCCGUGG
AAGAACUGACGAGUUCGGAUUCCCGACCGCAGCCCUGAGAGACGUCUCAGGGGCGGCCUCAAUGAUGAAUUAGAGGAAGUUAUUGUCGAGAAAACGAGACAGACUUUAGC
UUCCGUUGCAUCAGGACCAAAACACACCCAGCCCCCGCAAGGAGGGGUGUGUGUUUCACGCCCCUGGGAGUUAACGAUUCUCCGGCCCCACCAGUGUCAGACCACGCCAU
GGCGUGCCACUCUGCGGAGAGUGCAGUUUGUGACAGUGCCCCAGGAGGACUGGGUUUAAACAAAUUUUCUUAAAAUUUCUGAGGUUUGUUUAUUUCUUUUAUCAGUAAAU
AUGAUCAACCAACGGAAAAAGACGGGUCGACCGUCUUUCAAUAUG
>0.15-1
GGGUCUCUCUAGUUAGACCAGAUCUGAGCCUGGGAGCUCUCUGGCUAGCUGAGAACCCACGGGAAACGGGAGAGCGGCAGCGCCCGCCGAAACAACCGCAGGGAAACUGG
GUGAGGGGCCAGCCCCUGGCCCCGAAACAAUGUUAUGGGGCACUGGGAAAGACCAGAGAUCCUGCUGUCUCCCUCAGCAUCAUUCCAGGCACAGAACGCCAGAAAAUGGAU
ACAGCGGGGGAGACAUAUAUCACAGCCUGUCUCAUGCCCGACCCCGCUGGAUUCCCUGACAACUACGAGCACCUGCAUGAAGCGGAUGGCUUCAUUUGGUGACCCCGAUG
UGAUAGUUAGGGAAUCACUCCCCUGUGAGGAACUACUGUCUUCACGCAGAAAGCGUCUAGCCAUGGCGUUAGUAUGAGUGUCGUGCAGCCUCCAGGAGCCCCCCUCCCGA
GUAGCGAGAUCUGGGCGUCGGAACCGGUGAGUACACCGGAAUUGCCAGGACGACCGGGUCCUUUCUUGGAUAAACCCGCUCAAUGCCUGGAGAUUUGGGCGUGCCCCCGC
AAGACUGCUAGCCGAGUAGUGUUGGGUCGCGAAAGGCCUUGUGGUACUGCCUGAUAGGGUGCUUGCGAGUGCCCCGGGAGGUCUCGUAGACCGUGCACAACCAAGACACG
GUUGAUCUCACCCUUCGGGGGGCUAUAGAGAUCGCUGGAAGCACUACCGGACAACCGGAACAUUGCAGCAAUGCAGGGGAAGCUACUGUCCCAGGGAGGAAGGGCCGCCA
UCUGUGGCAAAUACCUCUUCAAUUGGGCAGUAAGGCUGUGCCUACUCAUACUUACAGUAGGGGUAGGCAUCUUGUAACGGGGCAUGGCUUAACUAAGGGGACUAUGGCAU
GUAUAGGCGCAAAGCGGGGCUUCGGUUGUACGGGGUUAGGAGUCCCCUCAGGUGCCAAGGUAUCUUUACGUCAACCUCGCUGUUGUUCCUUGUGACUGUACCUUUGGUAU
GAGUUCUCUAUCUCUAAAAUCGUUCCAGGACAUACUGGUGAGGAUGUCAAAAAUGCAAUUGGCAUCCCCAAUAGGUACAGUGUUGAUAGGACCUACACCUUCAAUAUUGG
AAGAAAUUUGAUGACUAAGUUGGUUGCACUUUAAAUUUUCCAUUAGUCUAUUAACACUGUACCAGUUUUGAAAGGGGUCUCCUAGAGAGCUUGCCUAGGGCCUUAUAACC
CGACUUGCUGAGCUUCUAUAGGAAAAAACCCUUUCCCAGCCUUGGGGUGGCUGGUCAAUAAAAACCCCCAUGUAAGUGGACUCUUAGUUACUCUGCUCAGCCCACCACUC
CACCACACCCAGGCACACACUACACACAGCCACCCGUCUCAGUUAAAACAGCCUGUGGGUUGCUCCCACCCACAGGGCCCACUGGGCGCUAGCACACUGAUUUUACGGAA
UCUUUGUGCGCCUGUUUUAUAUGGGAUCGGCUUGCUGAAAGUGCCUCGGCAAGAGGCGAGAGAGUUGUUAGUCUGUGUGGACCGACAAGGACAGUUCCAAAUCGGAAGCU
UGCUUAACACAGUUCUAACAGUUUUUUAGGGAGAGUUUGGCUUUCCCACAAGGGAAGGCCAGGGAAUUUCCUCCGCGAGUAAGACAGACUCUUCAGUCUGAGUUUGUGGA
GAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUAUUGCGUGGUGCCUUUUUUAGUGCUAGCAGGCGCCGCAAGCGUCGGCGCCUACGAACACGCGACCACGAUGCCUUUCC
CUGAAGAGACGAAGUGAUCAAGAGAUUGAAGACGAGUGAACUAAUUUUUUUUUAAUAAAAGUUCACCACGACUCCUCCUUCUCUCACAAGUCGAAACUCAGAGUCGGAAA
GGGAAUUUAAUUCGCUUUCGAUUCAAGGAUGAUGAUGACGGAGCAAGAAUAGAAAGCGCUUAUGUGACCGCCUCCAGGCCCUAAUAGACCUGGUCUCCAAGGCACUGGAG
GCGUUCAGACAGGAGUAAUCCCAGCUGAAACAGGGUUUUCACCCUUCCUUUCUUCUGGUUUCCUUCCUCUGCUUCAGCGGCCGAAGCCGCGCCCACGCCGAGUAGGAUCG
AGGGUACAGUCAUAGGACGCUCUAAUACGGACAUGGUGCGAAGAGUCUAUUGAGCUAGUUGGUAGUAGUCCGGCCCCUGAAUGCGGCUAAUCCUAACUGCGGAGCGCGCG
CCCCCAUUCCAGGAGGUAGCGCGUCGUAACGGGCAACUCUGCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCUUUUAUCUUUAUUCUGGCUGCUUAUGGUGACAAU
UGACAGAUUGUUACCAUAUAGCUAUUGGAUUGGCCAUCCGGUCAUUGGCCCCUGGGGAAAGAAGCCCCGCAAUUUCCCCGUGGCCCAAGGCUCGACUCCGUGGCCUGGAG
GCUAAGCGAACGGGUUGGGCUGCGCGUGUACCCCGGUUCGAAUCUCGAAUCAGGCUGGAGCCGCAGCUAACGUGGUAUUGGCACUCCCGUCUGACCCAAGCUGCACCAAC
CCUCCAGGAUACGGAGGCGGGUCGUUAUGGGCGGCGCAUGAGAGAAGCCCAAGAAUAAAACUACCCAUCAUGGAGAAGGUUCACGUUGAUAUCGAGGAAGACAGCCCCUU
CCUCAGAGCUUUACAACGGAGCUUCCCCCCAGUUUGAGGUAGAAGCCAAGCAGGUUACAGACAACGACCAUGCUAACGCUAGAGCGUUUUCGCAUCUAGCAUCUAAACUGA
UCGAGACGGAGGUGGAACCAUCCGAUACGAUCUUAGACAUUGGAAGUGCGCCCGCUCGCAGAAUAGAAAAACCAAAAGAAACACCAACCGCCGCCCACAGGACGUCAAGU

UCCCGGGUGGUGGUCAGAUCGUCGGUGGAGUUUACUUGUUGCCGCGCAGGGGCCCUCGUUUGGGUGUGCGCGCGACGAGGAAAACUUCUGAACGGUCCCAGCCUAGGUCG
GGUCGGCAUGGCAUCUCCACCUCCCCGUGGUCCGACCUGGGCAUCCGAAGGAGGACAGACGUCCACUCGGAUGGCUAAGGGAGUGCCAGCGGUUUGUGAGGAGGUUGGGG
UAGCCCACCGACUUGGGUAUGAUGGGUUUCUGUCAUACUACACGGUGCGAAACUCCGUACUUACACACGAGCCCACGGCCCAGCCUGGCUGAAGCUGUAGGUCAGGGGAA
GGACUAGAGGUUAGUGGAGACCCCGUGCCGCAAAGUGGCCACCCACACCACAUACGGAAGAAUGACUGCUGGAAUGGGGACUCUUCCUGACCUUGGGGGAAGCCAUGAUC
CAGGCGCCAGGGGGAAAGAAAAAAUAUAAAUUAAAGCACAUAGUCUGGGCAAGCAGGGAGCUAGAACGAUUCGCAGUUAACCCUAGCCUGACUAGCGGAUGCUAGAAGAA
AAAGUGGUUAGACUCCUUCUAAAAUUAGAGCAUAAUUAACAAAUUAUAAUUGGCUUAACCCUACCGCAUGAACCGAACUUGAUAAAAGUGCGGUAGGGGUAAAUUCUCCG
CAUUCGGUGCGGGCCCCAGUCCCCAGGUCAGGGUAUGACCCCGUGGUGUUGAAAGGACAGAAAACUCACCGCAACGAAACGAAACAAAUCGUUCAGAAACACUAGACCAU
GAGGGCCCCCCUAUAGUCCCGCUGAGGUGCUGCAGGCCCCGUGCGAUAGGCUAACUGUGGUGUUCCCCGCACUCCGUCGAGCGGUUAAUACGACGCUUACCAAGACAACA
AACGUUUUACCGGUUUGUAAAUUGGAAUUCUCGAAUUCGCCAGUCGCCUUCUUCGUAUCAAGGAUCUGGAAUUUGAACGGUAUCUUUAUCAUUUUAAUUUCUACUUUUCG
UACCAGUCACUAUUUUCUGCAUUCGCCCUUGCGUGUCCAUUGUUGCAAGGAGCGAUUUGGAGAAAAUAAACUCUGGUAACUAGAGAUCCCUCAGACCAUUUUAGUCAGUGU
GAAAAAUCUCUAGCAGUGGCGCCCGACAGGGACCGGAAAGCGAAAGAGAAACCAGAGCAUUAACAACUAUAUCCAGUUCAAGAGCAAACACCGUAUUGAACCAGUAUGUA
UGUUGAUAAGUGUUUUUCUCUCCACUUAAAUCGAAGAGAAGCAGGUUUCCCCAACUGACACAAACCGUGCAACUUGAAACUCCGCCUGGUCUUUCCAGGUCUAGAGGGGU
GACACUUUGUACUGUGUUUGGCUCCACGUUCGAUCCACUGGCGAGUGUUAGUAACAACACUGCUACUUCGUAGCGGAGCAUGACGGCCGUGGGAACUCCUCCUUGGUAAC
AAGGACCCACGGGGCCGAAAGCCACGUCCACAUGGACCCGUCAUGUGUGCAACCCCAGCACGGUAGCUUUGUUGUGAAACCCACUUUAAAGUGACAUUGAUACUGGUACC
CAAAUACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACAAGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUAUAAAAGCGCCCAGUUUAAAAA
GCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCUUUAAUCACAUUGAAAUUAAAAGGCGCGGGAUAGCGUGCGCUACCGGAUGGCGGGUAACUUAGGAAGC
GUUUCUUGAGCUUCCCUGGGAUGAGCGUUUGGGAGAGCUGAUUCUGCAGCCCAGAGAGUAGUCUCAGGGCAUCCUCUGGAGCCUGACCUGUGACGUCGCAUCAUAGACCG
CCAGUAGACCUGGGAGCAGAUUCACCGCCGCGGCCGUCUCCUUUAAGUGUGAAUCGUCUGUACUAGUUAGCUAACUAGCUCUGUAUCUGGCGGACCCGUGGUGGAACUGA
CGAGUUCUGAACACCCGGCCGCAACCCAGGGAGACGUCCCAGGGACGGAUUGGGUGAUGAAUUAGAAGAAGUUAUUGUCGAGAAAACGAAACAGACACUAGCUUCAGUAU
CAUCAGGACCAAAACACACCCAGCCCCCGCAAGGAGGGACGUGCUUUUCACGUCUCUGGGAGUUGACGACUCUCCGGCCCCACCAGUGUCAGACCACACCUGAUGUGCCA
CUCUGCGGAGAGUGCAGUCUCCGAUAGUGCCCCAGGUGGACUGGGUUUAAACAAACCUUCUUAAAAUUUCUGAGAUUUGUUUAUUUUGCAUAUUCAGUAAAUAUGAACAA
CCAACGGAAGAAGACGGGAAAACCGUCUAUCAAUAUG
>0.2-0
GGGUGGUGUGUACCACCCCUGAUGAGUCCGAAAGGACGAAACGCAGGAAUCUCAGACCAACCCAACCCGUGCAACUGCAAGUUUCACCCGGUCUUUCCGGGUCUUGAGAG
ACAAACAGAUGUACUGAGAUCAACUCCACGAUUGGUCUACUAGCGGGUACUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGUGGGAGCUCCCCCAUGGUA
ACAUGGACCCACCGGGCCAAAAGCCACGCCUCACAGCCUCAUGUGUGUGCAACCCCAGCACGGCAAUUUGUUUGUGAACCACACCUUAAGGUAACACUGAGACUGGUACU
UGGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGAGACACUCGGGAUCUGAGAAGGGGACCAGGAGUUCUAUCAAACUGCCCGGUUUAAAAA
GCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCCUUUAUCUAAACUCAAUUGAAAGUCAGGCCGGGAAGUUCCCGCCACCGGAAGUUGAGUAGACGGUGCU
GCCUGCGACUCAACCCCAAGGAGGACUGGGUACACCACUCGCAAAUUCGGAUCCUGGGAAACAGGCAGAACUUCGGUUCGUAAGCUCGGGUAAACUAGUCCUUUACCGUC
GUAUCGUACUUUGUGUGGCCGGUCUCCGAGGAGGUCGGGGUUGCCCACCGUUUGAUACUACAACGGUGCGAAGCUCCGUACUUACACGCGGGCUAUGACUAAAAAACCAG
GAGGGCCCGGGAAAAACCGGGCUAUCAAUAUGGGCCGACAGGAGUAAUUCCCGCCGAAACAGGGUUUUCACCCCUCCUUUCUUCGGGUGUCCUUCCUCGCGCCCGCAGGA
CCUGGAGAUAGACUGGGAGAUCUUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUAGCUGGUGGUGAGGAAGAACACAGGAUCUAGAAAAACC
AAAAGAAACACCAACCGCCGCCCACAGGACGUCAAGUUCCCGGGCGGUGGUCAGAUCGUCGGUGGAGUGUACUUGCUGCCGCGCAGGGGCCCCCGGUUGGGUGUGCGCGC
AACAAGGAAGACUUCAGAACGAUCCCAGCCCAGAAUCCCUGUGAGGAACUUCUGUCUUCACGCGGAAAGCGCCUAGCCAUGGCGUUAGUACGAGUGUCGUGCAGCCUCCA
GGACCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCGCUGGGGUGACCGGGUCCUUUCUUGGAAUAACCCGCUCAAUACCCAGAAA
UUUGGGCGUGCCCCCGCGAGAUCACUAGCCGAGUAGUGUUGGGUGGCCUCAAUGAUGAAUUAGAGGAAGUUAUUGUCGAGAAAACGAGACAGACUUUAGCUUCCGUUGCA
UCAGGACCAAAACACACCCAUGGGGGCCCAAGGUGAGAUGAAGCUGUAGUCUCACUGGAAGGACUAGAGGUUAGAGGAGACCCCCCCGAAAUAUUUGGACCUUGGGGAAA
GAAGCCUCGCAACUUCCCCAUGGCCCAGAUGGGCGGCGCAUGAGAGAAGCCCAGACCAAUUACCUACCCAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGACAGCCC
AUUCCUCAGAGCUUUGCAGCGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACUGAUAAUGACCAUGCUAAUGCCAGAGCGUUUUCGCAUCUGGCUUCAAAAC
UGAUCGAAACGGAGGUGGACCCAUCCGACACGAUCCUUGACAUUGGAAGUGCGCCCGCCCGCAGAAUCCAGUGUCAGACCACGCCAUGGCGUGCCACUCUGCGGAGAGUG
CAGUUUGUGACAGUGCCCCAGGAGGACUGGGUGGAAUUUAAUUCGUUAUCGGUUUGAAGGUGGGUAUGGCAGAGCAAGAAUAGAAAGCGCUUAUGUGACCGGCCAGGGGG
AAGAAAAAAAUAUAGGCUGAAACAUUUAGUAUGGGCAAGCAGAGAGUUAGAAAGAUUCGCACUUAACCCUAGCUAAGCUACUGUCCCAAGGGGGGAGGGCUGCCACUUGU
GAAUACCUCUUCAAUUGGGCGGUAAAGCUAUCGUAGUGUGUGCGAUAGCAUUGCGUUAGGGGUUACUUUCGAGAGAUUUGGAGGGGUUGGUCCCUCCGGUGAGAGAAGAA
AACUCAUCCCAAACAAAAAACAGGGUUUCUGAUAACCGGUGCGACAAUAUGCCCUUCCAAGGCACUGACCGGCACGGAGAAAAACUAUAAUGCCAUGUCGUAUAACGGCU
AGGCACCCCGAAAUCCUCCAGGACACCUCGACCUAGGUUGUGUAUCCGGUGAUUUGCGUACUGUUGCUUUGCGUAGAGGAUUUCUCUCCUGUUCACUACAUAUCAGGGGU
UUGAAGGCAGGGUCUACCUUUCGGGGGGAUAAAUUGUAACUUCCUGCAAAUAAGCAAAAUGCUCUGGUAACUAGAGAUCCCUCAGACCCUUUUAGUCAGUGUGAAAAAUC
UCUAGCAGUGGCGCCCAACGUGGGGCCGGAAAGCGAAAGAGAAACCGGAGGGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCCUGAGGAAGAGGUGGACGGCCAG
AUAAGUGGCUAGACUCUUUUUAAAUUAGAGCACAAUUUGGUUUAAUUCCAAUUGGCUUAACCCUACCACACUCACCGAACUAGAUAACGGUGUGGUAGGGGUAAAUUCUC
CGCAUUCGGUGCGGCCCCAGUCCCCAAGGUCAUGGGAAAAACUGACCCCGAAGUGCUGCAAGGGCAAGCGAGUAAGACAGACUCUUCAGUCUGAGUUGGUGGAUGUGAGU
GUACUUUGUUUGAGCAUACAGGUUAAUGUUAUCUUCAGCCGAACAGACAGCCUAUGCUCAAGCCAUUACUGGAGCCCACAAGGUAAUCUAGGUGGUAAUUGCGCGGGUAA
UGUCAUUGGUAAUGAUGCUCUAAAAGGUAAAACCCCUCGCCUACGUAAGCGUUAUUACGCCCGAACUACAGUUCCGCUCCUCUAGCACACAGAGGUCAAUUGGGUGCGAC
CCCCCCCUCUCCCGUGGGUCAACGGGAACCCAUGGGACGCUAGUUGUGAACAAGGUGUGAAGAGCCUAUUGAGCUACUUGAGAGUCCUCCGGCCCCUGAAUGCGGCUAAU
CCUAACCACGGAGCAAGUGCUCUCAAACCAGAGAGUGGCUUGUCGUAACGCGUAAGUCUGUGGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCUUUUAUUUUUAUCAU
GGCUGCUUAUGGUGACAAUCUAAGAUUGUUAUCAUAUAGCUUUUGGAUUGGCCAUCCGGUUGCCGGCGGGGGAGACAUUUAUCACAGCAUGUCUCAUGCCCGACCCCGCU
AUCGUAUUGUUCGGUCUACAUAAUGACGAGUCAGGUCAGGCUCCGCACUAGGUUCGGUCACCUAGGGGAUGGAGACAUGGAAGGGAUCUCGUGCGCUAUAAGUCUAACGA
AAGUUGAGCUUGCGACAUGGGUCUAUGCCUGGAUAAGUUGUGUAGGUUGCCGCAUACGGGACUCGGCUUGCUGCGGUGCACACAGCAAGAGGCGAGACGCCCACGCUAGG
GGGUAGAGUAGACUCAUCUCCGGGUGUUGACGACACCCCGACCCCAAGUUAUGGUGGGCUGGGCGGGGGCAUCGAAAGAUGCGCCGGUUUCCUUGAUCACCGGUUACGCC
AACCCCGUUCAGUUCACCACCAGCAGGACCUACGCUGCCCUAGAGGUUUUGCUAGGGAGGAGACGUGUGUGGCUGUAGCCACCCGUCCCGGGUACAAGUCCCGGGUGGUG
AGGACGGUGUCUGUGGUUGUCUUCCCAGACUCUGCUUUCUGCCGUCUUCGGUCAAGUACCAGCUGGUGGUCCGCAUGUUUAGGUCUCUCUAAGUAGACCAGAUCUGAGCC
UAGGAGCUCUCUGGCUAACUAAGGAACCCACUUCUGUACUAGUUAGCUAACUAGGUCUGUAUCUGGCGGCUCCGUGGAAGAACUGACGAGUUCGUAUUCCCGACCGCAGC
CCUGGGAGACGUCUCAGGGGCUCGGGUCGGCAUGGCAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUCCGUAGGAGGACAGACGUCCUACUCGGAUGGCUAAGGGAGAG
CCAGUAGUGGCCACCCACACCACAUACGGAAGAAUGACUGCUGGAAUGGGGACUCUUGAUGAACAACUACAUGCAGUUCAAGAGCAAACACCGUAUUGAACCCGUAUGUU

UGAUCAUUAUAAGUACAGUAUUAGUGGGACCUACACCUGUCAGGUAGUUGGAAGAAAUCUAAUGUCUCAACUUGGUUGCACUUUAAAUUUUCAUGUCCAUUGAACGUCCA
GUGGCUCGACUCCGUGGUCUGGGGGAACGUGAACGGGUUGGGUCGCGGUGUACCCCGGUUCGAGUCCAAAGCUAAGCGAUCACGCUCGGAUCGGCCGGAGCCGCGGCUAA
CGUGGUAUUGGCUAUCCCGUCUCGACCCAGCCGACGAAUAUCCAGGGUACGGAGUAGAGUCGUUAGUAGGACUAGCAAAACGGAGGGACUAGCCAUAGUGGCGAGCUCCC
UGGGUGGUCUAAGUCCUGAGUACAGGACAGUCGUCAGUAGUUCGACGUGAGCAGAAGCCCACCUCGAGAUGCUACGUGGACGAGGGCAUGCCCAAGACACACCUUAACCC
UAGCGGGGGUCGCUAGGGUGAAAUCACGCCACGUGAUGGGAGUACGACCUGAUAGGGCGCCGCAGAGGCCCACUAUUAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCA
CAUGGAGUUUUAAAACAGCCUGUGGGUUGUUCCCACCCACAGGGCCCAUUGGGCGCUAGCACACUGGUAUCACGGUACCCUUGUGCGCCUGUUUUAUCCGAGACAAUUAA
GCCUGUUCUUGACCUUACUAGCUCCAUCAUAUAUCGGGUGGAAUGAAUAACAUGUUCUUUUGCUCUGCGCAUGGAGAUGCCACCAUGCGCUCUCGGGCUCUUCUGUUUCU
GCUCCUCGUGUUUUUGCCUAUGCUGCCCGCUGUAACGGGGCAUGGCUUAACUAAGGGGACUAUGGCAUGUAUAGGCGCAAAGCGGGGUUACGGUACGCGACUUAGGAGUC
CCCUUAGGUUUUUUAGGGAAAAUUUGGCCUUCCAGCAAGGGAAGGCCGGGAAAUUUUCUU
>0.2-1
UGUCCGUAGUGGAUGUGUAUCCACUCUGAUGAGUCCGAAAGGACGAAACGGAUGACGCAGGUUUCCUCAACUGACACAACCGUGCAAACAUGAAGUUUCGCCCGGUCUUU
CCGGGUCUUGAGAGACGACAUUUGUACUGAGAUCAACUCCACAAUUGGUCUACUGGCGGGUGCUAGUAAUGACUUUGUUGCUUCGUAGCGGAGCACGCAAGCCGCGGGAA
CUCCCCCUUGGUAACAAGGACCCGCGGGGCCGAAAGCCACGUUCUCUGAACCUUGCGUGUGUGCAACCCCAGCACGGUAAUACUGUCAUGAAACACACUUUAAGGUAACA
CUGAGACUGGUACUUGGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACAAGCGACACUCGGGAUCUGAGAAGGGGAUUGGGGCUUCUAUCCAAGC
GCCCAAUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGCCGGCACCUUUUCCUUUACUCACAACUUGUUGAGAGUCAGGCCGGAUUACAUGUCCGCCACCGGAUGU
UGGAUGACGGUGCUGCCUGCAGUCCAACCCCAGGAGGACUGGGUACACCACUAGCACAAAUCGGAUCCUGGGAAACAGGCAGAACUACGGUUCGUAAGCUCGGGUAGGCC
GUCAACCUACCGCCGUAUCGUAUUGUGUUUGGCCGGUUUGUGAGAAGGUUGGAGUUGCCCACCGGUUGGGGUAUGACGGUUUCCUGUCAUACUACAGCGGCGCGAAGCUC
CGUACUUACACACGAGCUAUGUCUAAAAAACCAGGAGGACCCGGGAAGCCCCGGGUCGUCAAUAUGUUGAGACAGGAGUAAUCCCAGCUGAAACAGGGUUUUCACCCUUC
CUUUCUUCUGGUUUCCUUCCUCUCGCCGGAAGGUCCUGGGAAAAGACUAGGAGAUCUUCUGCUCUAUUCCAACAUCAGUCACAAGGCACCGAGCGCCGAACACUGUGACU
GAUGGGGGAGAAGACCACAGGAUCUAGAAAAACCAAACGUAACACCAACCGCCGCCCAAUGGACGUUAAGUUCCCGGGUGGCAGCCAGAUCGUUGGCGGAGUUUACUUGU
UGCCGCGUAGGGGCCCUAGAUUGGGUGUGCGCGCGACUAGGAAGACUUCGGAGCGGUCGCAACCUCGUUUUAAGUAGAAACUGUCUAGCCAUGGCGUUAGUAUGAGUGUC
GGGCAGCCUGCCAAGGACUCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUUGCCAGGACGACCGGGUCCUUUCUUGGAUCAAACCCGC
UCAAUGCCUGGAGAUUUGGGCGUGCCCCCGCGAGACUGCUAGCCGAGUAGUGUUGGGUCGCGAAAGGCCUUGUGGUACUGCCUGAUAGGGUGCUCGAGAGUGGUCUUAAU
GAUGAAUUAGAAGAGGUCAUUGUCGAGAAAACGAGACAGACAUUGGCCUCUGUUUCUUCUGGACCUAAACAUACACACGGGGCCCGAGCUCUGAGGGAAGCUGUACCUCC
UUGCAAAGGACUAGAGGUUAGAGGAGACCCCCCGCAAAUAUUGGGCCCAUGGGGAAAGAAGCCCCGCAAUUUCCCUGUGGCCCAAAUGGGCAAUACGUAAGAGUAGCCCA
AAAACUGAAUACACCUACCCAUCAUGGAGAAAGUUCACGUUGACAUCGAGGAAGACAGCCCGUUCCUUCGAGCUUUACAGCGGAGCUUCCCACAGUUUGAGGUAGAAGCC
AAGCAGGUCACCGAUAAUGACCAUGCUAAUGCCAGAGCGUUUUCGCAUCUGGCCUCCAAAUUGAUCGAGACGGAGGUAGAACCAUCCGAUACGAUCCUAGACAUAGGCAG
UGCACCGGCACGCCGAAUCCAAUGUCAGACCACAUUUCGAUGUGCCACUCUGCGGAGAGUGCUGUCUGCGAUAGUGCCCCAGGUGGACUGGGUGGCGUUUAAUUCGCUAU
CAAUUUGAGGAUGAUGAUGGCUUAGCAAGAAUAGAAAGCGCUUAUGUGACCGGCCAGGGGGAAAGAAAAAAUAUAGAUUAAAACAUCUAGUAUGGGCAAGCAGGGAACUA
GAGAGAUUUGCAGUCAACCCUGGCCAAGUUGCUGUCCCAGAGGGGGAGGGCCGCCAUUUGUGGCAAGUACCUCUUCAACUGGGCAGUAAAAUAGACAGGGUGUUUGCGAU
AGCAUCCUGUUCUAGGUUGGUUUCUUAGCGAUUAUGGAGGGGUUGGACCCUCCUGGUGUGAGAAGGAAACACAUCAAAACAAAAAACAGGGUUCUGUAACCGGUGCGACA
AUAUGCCCUUCCAAGGCACUGACCGGCACGGAUAAAACUUUAUUUGCCUAGUCGUAUAACGGCUAGGCACCCCGAAAUUCCCCAGGAUUUCUCGACCUAGUUCGUUGUUA
UCUGGUGACUUGCGCUAACUGUUGCUUUGCGUAGAGAAUUUCUCUCCUUUGUUAAUUCUUUGACUUAGGGGUUUGAAGGUUGGGUCUACCUUUCGGGGGGAUAAAUUGUA
ACUUCCAACAAACAAGCAACAUGCUCUGGUAACUAGAGAUCCCUCAGACCAUUUUGGUAGCGUGGAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACCCGAAAACGAAAG
UAAGACCAGAGGGGCCUUCUGGUCGUGUUCUUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGACAGCCAAGAAAGUGGUUGGACUCCUUUUAAAAUUAGAGCGCAAUUA
GUCAAUCAUAAUUGGCUCAACCCUACCGCAUGAACCGAACUUGAUAAAAGUGCGGUAGGGGUAAAUUCUCCGUAUUCGGUGCGGAGGAGAAAACCCCAGUCCCCAGGUCA
GGGUAUGACCCCGUGGUGUUGAAAGGACAAGCGAGUAAGACAGACUCUUCAGUCUGAGUUCGCUGAAACGAGUGUAAAUCUGGCAUAGCAUACAGGUUAAUGUUAUCUUC
AGCCGUACAGGGAGCCUUGAACCGAGCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAUGAAGCUCUAAGGGUUAAACCCUU
CCGCCUACGUAAGCGUUAUUACGCCCUAAUUGUGGACAGUUCCACUCCCUUUAGCACACAAAGGUCCAAUCGGGUGCGACUCCCCCCCCUCCUGUGGGCUACAGGAACCA
UGAGACGCUAGACAUGAACAAGGUGUGAAGAGUUUAUUGAGCUACUAUAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACCAUGGAGCAAGUGCUCACAAACCAGU
GGGUUGCUUGUCGUAACGCGCAAGUCCGUGGCGGAACCGACUACUUUGGGUGUCCGUGUUUCACUUUUUACUUUUAUGACUGCUAAUGGUGACAAUUUAAUAUUGUUACC
AUUUGGCUUGUCAAAUUGAUCACAUAUACAGCGGGGGAGACAUUUAUCACAGCGUGUCUCAUGUCCGGCCCCGUUGGAGUAUGAUUAGGUCUACGUGAUGACGAGUCAGG
UCAGGCUCCGCACUGGGUUUGGUCGCCCAGGGGAUGGAGACACGGAAAGGACAUCGUGUGCUAUAAGUCUGACGUAAGUCGAGCUUGCAACAUGGGUCUAUGCCUGGAUA
AGUCACGUGGUUGCCCCUACUGGACUCGGCUUGCUGAAUUGCACUCGGCAAGAGGCGAAAGGCCCACGCAAGGGGAGAGAAGAUUUUCGCCUCCGGGUGUUGACGACACC
CCGACCCCAUAUUAUGGUGGGCUGUGUGGGGGCACCGAAAGGUGCGCCGGGUCCUUUUAGCCGGUUACGCCAACCCUGCACAGUUCACCACCAACGGGACCUACGCUGCC
CUAGCGGUUAUGCUGGGGAGGAGAUGAGUGUGACUUAAAUCACCCGUCCCGGGUACAAGUCCCGGGUGGUGAAGACGGGGUCUGGUGCUGCUAAAAGUUUGGACAGUCCC
CGGUUUCUGCUCUCUUCGGUUCAAACCAGCUGGUGGUCCGCCUGUUUGGGUCUUUCUGGUUAGACCGAAUCUGAGCCUGGGAGCUCUCUGGCUAGCUAGGGAAUCCACGU
CUGUACUAGUUGGCCGACUAGCUCUGUACCUGGCGGACCCGUGGUGGAACUGACGAGUUCGGAACACCCGGCCGCAACCCUGGGAGACGUCCCGGGGACUCGGGUCGGCA
UGGGAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUCCGUAAGGAGAGCGUAGCUCCCUCGGAUGGCAAAGGGAGAUCCACCAGCGGCCAUCCUCACCACAUGCGGAAGA
ACGAGUGCUGGAACGGGGAGACUUGAUGAGUAACUACAUACAGUUCAAGUCCAAAUGCCGUAUUGAACCCGUAUGUCUGCUCUUGAUAGGUACAGUAUUAGUAGGACCGA
CACCUGCCGAAUAAUUGGACGAAAUAUGUUAACUCAGAUUGGUUGCACUCUAAAUUUCCAUGUCCAUUGAACGUCCAGUGGCUCGACUCCGUGGCCUGGAGGCUAAGCGA
ACGGGUUGGGCUGCGCGUGUACCCCGGUUCGAGUCCCUGCUCGAAUCAGGCUGGAGCCGCAGCUAACGUGGUACUGGCACUCCCGUCUCGACCCAAGCCUGCUAACGAAA
CCUCCAGGAUACGGAGGCGGGUCGUUAGUAGGACUAGCAGACGGGAGGACUAGCCAUCGUGGUGAGAUCCCUGAGCAGUCUAAAUCCUGAGUACAGGAUAGUCGUCAGUA
GUUCAACGCAGGCACGGUUCUGCCUUGAGAUGCUACGUGGACGAGGGCAUGCCCAAGACUUGCUUUAAUCUCGGCGGGGGUCGCCGAGGUGAAAACACCUAACGGUGUUG
GGGUUACAGCCUGAUAGGGUGCUGCAGAGGCCCACGAAUAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUUUCCCACCAACAGGGCCCACGGGGCGCCAG
CACUCUGGUAUCACGGUACCCUUGUGCGCCUGUUUUAUAAGAGACUAUUAAACCUGUGCUUGACUUAACAAACUCUAUCAUACAACGGGUGGAAUGAAUAACAUGUCUUG
UGCUUCACCCAUGGGACCACCAUGCGCCCUAGGGCUAUUUUGUUGUUGUUCUUCGUGCUUCUGCCUAUGCUGCCCGCUGUAACGGGACAAGGCUUGACUGAGGGGACUGU
AGCAUGUAUAGGCGCUGUGCGGGGCUUCGGUUGUACGCGGAUAGGAAUCCCCUCAGGUUUUUUAAAGAGAAUUUAGCCUUCCAGCAAGAAGAGGCCAGGAAAUUUUCCU
>0.25-0
UAGUGAGACUAGGUUCGUGUGAUGACGAGUCAGGUCGGGCUCCGCGCUGGGUUUGGUCACCCAGGGGAUGGAGAUAUGGAAAGGAUCUCGUGCGCUAUAAGUCUAACGAA
AGUUGAGCUUGCAACAUGGGUCUAUGCCUGAAUAAGUCACGUGAGUUGCCACCACUAUGGCGCCCAACGUGGCGUUGGAUACGAGGGAAUUUCGUGAGGAAGACGACGCG
GUUUGCCGGCCCGGAUUAAAAGAGAAACGAAAGUAAACUUUCUUCGGCCGCCGCGGGAGCCUGCCGCGUAGGACCUGAAAGUAAGUGGUGCGCUCGGAUAUGGGGCAGGA

AUUAAGCCAGCACGAACGUUAUGUGGAACAAUUAAAACAGGCUUUAAAGACACGGGGAGUAAAGGUUAAAUAUGCUGAUCUCUUAAAGUUUUUUGAUUUUGUAAAGGAUA
CUUGUCCUUGAUGAGCAACUACAUACAGUUCAAGUCCAAAUGCCGUAUUGAACCUGUCUGUCUUUUGCUCUACAACGGGGGAGACAUAUAUCACAGCCUGUCUCGUGCCC
GACCCCGUUGGGCCUCCAGGCCUUGCAACACUUGGUCCGGACGGCCCUGGAGGCAGGGGGCCACGUGUGGUGCCGUCCGCGCCCCCUAUGCUGUAAGCGAAGCACCACCU
CUGGUAACUAGAGAUCCCUCAGACCCUUUUAGUCAGUGUGAAAAAUCUCUAGCAGUGGCGCCCAACGUGGGGCCGGAAAGCGAAAGAGAAACCGGAGGUGUCUUGGAGCG
CGCGGAGUAAACAUAUAUGGUUCAUAUAUGUCCGUAGGCACGUAAAAAAAGCGAGAAAUCCUCCAGGACACCUCGACCUAGGUUGUGUAUCCGGUGAUUUGCGUACUGUU
GCUUUGCGUAGAGGAUUUCUCUCCUGUUCACUACAUAUCAGGGGUUUGAAGGCAGGGUCUACCUUUCGGGGGGAUAAAUUGUAACUUCCUGCAAAUAAGCAAAAUGAUGC
AGGUUUCCCCAACUGACACAAACCGUGCAACUUGAAGCUCCGCCUGGUCUUUCCAGGUCUAGAGGGGUGACAUUUUGUACUGUGCUUGGCUCCACGCUCGAUCCAUCGGC
GGGUGUUAGUAACAGCACUGUUGCUUCGUAGCGGAGCAUGAAGGCCGUGGGACCUCCUCCUUGGUAACAAGGACCCACGGGGCCAAAAGCCACGUCCACUGGACCCUUCA
UGCGUGCAUUCCCAGCACGGCAACUUUACUGUGAAAUCCACUUUAAGAUGACAUUGAUACUGGUACCCAAGUGCUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGA
GGUAACACGUGACACUCGGAAUCUGAGAAGGGGACCGGGGUCUCUAUAAAAUCGCCCGGUUUAAAAAGCUUCUACGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCC
UUUAUACCUACUGACUUAGCCAGGGGGAAAGAAAAAAUAUAGAAUGAAGCAUCUAGUAUGGGCAAGCAGGGAGUUGGAAAGAUUCGCAGUUGAUCCUAGCCUAAACGUUU
CACCGGUUUGUAAAUUGGAACUUCGUAGUUCGCCAGACGCCUUCUUCGUAUCAAGGAACUGGAAAUUGUACGGUGAGUCUGUCAUUUUAUUUUCUACUUUUCGUACCAGU
CUGUAACGGGGCAUGGCUUAACUAAGGGGACUAUGGCAUGUAUAGGCGCAAAGCGGGGUUACGGUACGCGACUUAGGAGUCCCCUUAGGCAACGUCAGGCCACAAUUCUG
UGCCACUCCGCUGGGGAGUGCGGCCUGCGUAGCCCCAGGAGGACUGGGUAGGCCUUCUGGUGAUGUUUCUGGCCACCCAGGAGGUCUUGAGGAAGAGGUGGACGGCCAGA
CAGGACCUACGCUGCCCUAGAGGUUUUGCUAGGGAGGAGACGUGUGUGGCUGUAGCCACCCGUCCCGGGUACAAGUCCCGGGUGGUGAGGACGGUGUCUGUGGUUGUCUU
CCCAGACUCUGCUUUCUGCCGUCUUCGGUCAAGUACCAGCUGGUGGUCCGCAUGUUUAAGCUACUGUCCCAGGGGGGAAGGGCUGCCACUUGUGGCCGCUACCUCUUCAA
CUGGGCAGUAAAGUCGGCCGUAAACACCCAAUCAUCAGUAGGCCGAGAAGAGUCGCGCUAAGCGCACUGAUGAGUCUUUGAGAUAAGACGAAACUCUUCCUGGGAAUAGA
CUGGGAGAUCUUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUAGCUUGUGGUGAGGAAGAACACAGGAUCUUUUUUUAGGAAAGAUCCGGCC
UUCCCACAAGGAGAGGCCAAGGAAUUUUCUUAGGGUGCGAGAGCGUCAGUGCAACAGCGAAGCGGAACGGCGAAACACACCUUGUGUGGUAUAUUACCCGUUGUGGUGCC
UACUCCUACUGUCUGUGGGAGUAGGCAUCUAGAGAAAAGACAGACUUCAGUCUGAGUUUGUGGAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUAAUGUUAUCUUCAG
CCAUAUAGACAGCCUGGGAGUCAGCCCUUACUGGAGCCCACAAGGUAAUCCAGGGGGUAAUGGUAAACCCGGUAAUGUCGGUGGUAAUGAUGCUCUACCAGUGUCAGACC
ACACUUUAAUGUGCCACUCUGCGGAGAGUGCAGUCUGCGAUAGUGCCCCAGGUGGACUGGGUGUUGGUACAGUGUUAGUAGGACCAACACCUGUUAAAUUAUAGGAAGAA
ACUUUUUAACACAGAUUGGCUGUACUCUAAAUUUUCCAUUAAGUCCUAUUAAACUGUACCAGUUGGAGAUGGAUGCUGGAUCCUGGAAAACAGGUUUAACGGGUAGGGGG
UGGUGCCCCUACCGACGCAUCACUCCAGAUACCAAUGAACAACCAACGGAAAAAGACGGGUAAACCGUCUAUCAAUAUGGGUUUGUGAGGAGGUUGGGGUAGCCCACCGA
CUUGGGUAUGAUGGGUUUCUGUCAUACUACACGGUGCGAAACUCCGUACUUACACACGAGCCGUUAAGAUGUGAUCUUGCUUCCUUAUACAAUUUUGAGAGGUUAAUAAG
AAGGAAGUAGUGCUAUCUUAAUAAUUAGGUUAACUAUUUAGUUUUACUGUUCAGGAUGCCUAUUGGCAGCCCCAUAAUAUCCAGGACACCCUCUCUGCUUCUUAUAUGAU
UAGGUUGUCAUUUAGAAUAAGAAAAUAACCUGCUAACUUUCAAGGGUCUUUCUGGUUAGACCGAAUCUGAGCCUGGGAGCUCUCUGGCUAGCUAGGGAAUCCACAAGUGG
UUGGACUCCUUUUAAAAUUAGAGCGCAAUUAGUCAAUCAUAAUUGGCUCAACCCUACCGCAUGAACCGAACUUGAUAAAAGUGCGGUAGGGGUAAAUUCUCCGUAUUCGG
UGCGGAGGAGAAAAUUUCAUCGAGGCCACGCGGAGUACGAUCGAGGGUACAGUGAAUGGAGCUAUAAUCAUAAUUUCUGAAUUAGUCAUUGGUAAUUCAAUGCCAACCUC
CACUGGGUGGGUUAAGGUUGAGGUAUAGAAUCCUAUUCGCUCCUGAUAGGAGAAAUUCUAUAUUGCUUAUAUACGUGCUUAUGCACGUAUAUAAAUGCUCAUGCUAAAUU
GCAUGAAUGCCCCUAAGGGAUGCAGUAGGACUAGCAAAAGUAGGGGACUAGCGGUAGCAGUGAGUUCGUUGGAUGGCCGAACCCCUGAGUACAGGGGAGUCGUCAAUGGU
UCGACACUCCAUUAGUCGAGGAGUCUCGAGAUGCCAUGUGGACGAGGGCAUGCCCACGGCACAUCUUAACCCAUGCGGGGGUUGCAUGGGUGAAAGCGCUAUUCGUGGCG
UUAUGGACACAGCCUGAUAGGGUGUAGCAGAGACCUGCUAUUCCGCUAGUAAAAACUCUGCUGUACAUGGCACAUGGAGUUACAGGACGCUCUAAAUACAGACAUGGUGCG
AAGAGUCUAUUGAGCUAGUUAGUAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACUGCGGAGCACAUGCCCACAAACCAGUGGGUGGUGUGUCGUAACGGGUAACUCU
GCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCUAUUUAUUCCUAUACUGGCUGCUUAUGGUGACAAUUGAGAGAUUGUUGCCAUAUAGCUAUUGGAUUGGCCAUCCG
GUGCCCCAGUCCCCAAGGUCAUGGGAAAAACUGACCCCGAAGUGCUGCAAGGGCAGGCGAACGGUGAGUACGCCAACCCCGUAAGGAAGGACAAGGCUGUCCUUGAGUGC
CAACGACGCUCCGGCCCCAAGUGGCCACCCACACCACAUAAGGAAAAAUGACUGCUGGAACGGGGACUCCUGGACUCGGCUUGCUGAAGUGCACUCGGCAGGAGGCGAAA
GUCAGUCGCCAGUGCGAGGCCCAUUCUCUCUCAAUGGGUAUUGGGUGCAACCCCCCCGUCCAUCUCGAACGUCAUCGAGACCAUGGAUGGCGUUGUCUCGUAGAGGUCCA
GAUCUCUCCUGUUGGCAACUCGAAAUCUCUACGAGAUAACAGUUUGUUAAAACAGCUGUGGGGUUGUUCCCACCCCAGAGGCCCACGUGGCGGCUAGUACUCCGGUAUCA
CGGUACCCUUGUACGCCUGUUUUAUACCGGGGCCCAACACCAGGGGAAGCUGUACCUUGGUGGUAAGGACUAGAGGUUAGAGGAGACCCCCCGCACAACGUGGCCGGCAU
GGCCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACGAUCCGAGGGAGCUACUCCUCUCGAGAAUCGGCAAAUGGGGCCCCAUCCCUGUGAGGAACUUCUGUCUUCACGCG
GAAAGCGCCUAGCCAUGGCGUUAGUACGAGUGUCGUGCAGCCUCCAGGACCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCGCUG
GGGUGACCGGGUCCUUUCUUGGAAUAACCCGCUCAAUACCCAGAAAUUUGGGCGUGCCCCCGCGAGAUCACUAGCCGAGUAGUGUUGGGUGGUCUUAAUGAUGAAUUGGA
GGAAGUCAUUGUUGAGAAAACGAAACAGACAUUGGCUUCUAUCACUUCUGGACCCAAACAUACUCAGGCUCGCGCCCGUAGUCUGGAGAAUCAGUCGCCAGGGUUGCGUU
GCGGUAUGCCCCGGUUGGAGCCUAAGCGCGGCUCGUAUCGGCCGGUUUCCGCGACAAGCAGGGUUUGGCAGCCCCGUUAUUUCCAAGACCCCGCCAGCCGACUUCUCCAG
UUUACGGGAGCGAGCCCUU
>0.25-1
CGUAUUGUUCGGUCUACAUAAUGACGAGUCAGGUCAGGCUCCGCACUAGGUUCGGUCACCUAGGGGAUGGAGACAUGGAAGGGAUCUCGUGCGCUAUAAGUCUAACGAAA
GUUGAGCUUGCGACAUGGGUCUAUGCCUGGAUAAGUUGUGUAGGUUGCCGCAUACGUGGCGCCCAACGUGGGGCUGGAUACGAGGGAAUCCUGUGAGGAAGAGAGCGCGU
AACACGGCCGGCAGUCAAAGUGAAAGAAAAACCUUCCAGCUGCCGCGGGAGCCUGCCGCGUCGGCAAUAAAGGUAAGCGUUGCGCUCAAAUAUGGGACAAGAAUUAAGCC
AACAUGAACUCUAUGUAGAACAAUUAAAAAAGGCUCUAAAGACACGGGGAGUAAAGGUUAAGGGUAAUGAUUUGCUCAAAUUCUUUGAUUUUGUAAAGGAUACCUGCCCU
UGAUUAACAACUACGUACAGUUCAAGAGCAAACACCGUAUUGAACCAGUAUGUUUGUUGGUGGCUGGCGGGGGAGACAUUUAUCACAGCAUGUCUCAUGCCCGACCCCGC
UAUGCCUCCAGGCCUUAAAUGACCUGGUCUCCAAGGCCCUGGAGGCCGGGGGACACGUGUGGUGCCGUCCGCUCCCCCCAGGGAAUAACUGAUGCACCACCUCUGGUAUC
UAGAGAUCCCUCAGAUAACUGUAGUCUGUGCGGCCAAAAUCUCUAGCAGUGGCGCCCGAACAGGGACUUGAAAACGAAAGUAAGACCAGAGGUGUCUUGGAACGCGCGGG
UCAAAUAUAAGUGGUUCACUUAUAUCCGUAGGCACGAAAAAUUGCGUGAAAUUCCCCAGGAUUUCUCGACCUAGUACGUCUACCUGGUGACUUGCGCUGUCUGUUGCUUU
GCGUAGGGAAUUUCUCUCCAAUAUUAUUAUCUUUAAUUUGGGGUUUGAAGGUUGGGUCUACCCUUCGGGGGGAUAAAUUGUAACUUCCAACAAAUAAGCGACAUGACGCA
GGAAUCUUAGACCAACCCAAACCCGUGCAACUGCAAGUUUCGCCCGGUCUUUCCGGGUCUAGAGAGACAAACAGGUGUACUGAGACUGACUCCACGAUCGGUCUACUAGC
GGGUGUUAGUAACACUCAUUUUGCUUCGUAGCGGAGCACAUGAGCGGUGGGACCUCCCCCCAUGGUAACAUGGACUCACCGGGCCAAAAGCCACGCCUCACGGCCUCAUGU
GUGUGCAACCCCAGCACGGCAACUUGUUUGUGAAACCCACCUUAAGGUAACACUGAGACUGGUACUUGGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAG
GUAACACGAGACACUCGGGAUCUGAGAAGGGGACCAGGAGUUCUUUAAAACUGCCUGGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCCU
UUAUCAAUACUACAUUGGCCAGGGGGAAGUAAAAGAUAUAAGCUAAAACAUAUAGUAUGGGCAGGCAGGGAACUAGAACGAUUCGCGGUUAAUCCUGGCCGACAAACGUU
UUACCGGUUUGUAAAUAGGAGCACACAAGCUCGCUAGUCGCCCCCUUCGUAUCGGGGAUCUAGAGAUUAAACGGUAUACCUGUCAUUUUAAAUUUCUACUUUUCGUACCA

GUCUAUAACGGGGCGGGGCUUGGCCGAGGGGACUGUAACAUGUAUAGGCGAAAAGCGGGGCUGCGGUUAUACGCGGAUAGGAAUCCCCUCAGGCCGUGUCAGGCCACGUA
AGUGCCACUUCGCUGAGGAGUGCAGCCUGUACAGCCCCGGGAGGACCGGGUGGGCCUUCUGGUCGUGUUCUUGGCCACCCAGGAGGUCCUUCGCAAGAGGUGGACAGCCA
AGAAAGACCUACGCUGCCCUAGCGGUUCUGCUAGGGAGGAGAUUGGUGCGGCUUAUGCCGCCCGUCCCGGGUACAAGUCCCGGGAGGUGACGGUAGCGUCUGUAUUUGCC
CUGUAUCAGGUGUAUAGCUGCUCUCUCGCAGUCGCUGCUAUCAGCCGUCUCCGGUCCGAUGCCAGCUGGUGGUCUGCUCUUUUAGACUUCUGUCCAGAGGAGGCAGGGCU
GCCAUAUGUGGCAAGUACCUCUUCAACUGGGCAGUAAAAUCGGCCACAAACACUCAAUCCUCCGUUGGCCGGCAAAAGUCUGGGCUUAGCCACUGAUGAGCCGUUGAGAU
ACGGCGAAACUUUUGCUGGGAUAGACUAGGGGAUCUUCUGCUCUGCACAACCAGCCACACGGCACAGUGCGCCGAUAUAGGUGGCUGGUGGUGCUAGAACACAGGAUCUU
UUUUUAGGGAGAUUUUGGCCUUCCCAGAAGGGGAGGCCUGGGAAUUUCCUUUGGGUGCGAGAGCGUCUAUACAACAGCGAAGCGCGCCAGGGAAACACACCAUGUGUGGU
AUAUUAUCUGGCACUAUGCCUACUCCUACUCACAGUAGGGGUAGGCAUCUAGCGAGUAAGACAGGCUCUUCAGUCUGAGUUCGCGGAAACGAGUGUGAAUCUGACAACGU
AUGCAGGUUAAUGUUAUCUUCAGCCGUACAGGGAGCCUUGAACCGAGCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAUGA
AGCUCUACCCGUGUCAGGCCGCAAAGCGCCACUUCGCCAAGGAGUGCAGCCUGUAUGGCCCCAGGAGGACUGGGUGUAGGUUCAGUGUUGAUAGGACCUACGCCUGUCAA
GUAAUUGGAAGAAAUAUAAUGACUCAGCUUGGUUGUACUUUAAAUUUUCCCAUUAGUCCUAUUGAACUGUACCAGUGCGGUAAGCAAGACGGAUCCUGGGAAACAGGUUU
AACGGGCUGGGGGUGGUGCCCCGGCCGUCGCAUCACUUGCUGAUACAAAUGAAUAACCAACGGAAAAAGGCGAGAAGUACGCCUUUCAAUAUGGGUCUCCGAGGAGGUCG
GGGUUGCCCACCGUUUGACUGCAACGGUGCGAAGCUCCGUACUUACACGCGGGCUGCAAAAAUGUGAUCUUGCUUGUAAAUACAAUUUUGAGAGGUUAAUAAAUUACAAG
UAGUGCUAUUUUUGUAUUUAGGUUAGCUAUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGGAAGCCCUCUCUGCGGUUUUUCAGAUUAGGUAG
UCGAAAAACCUAAGAAAUUUACCUGCUACAUUUCAAAGGUCUCUCUUAGUAGACCAGAUCGAGCCUAGGAGCUCUCUAGCUAGCAAAAGAACCCACAAGUGGCUGGAUUC
CUUUUAAGUUAGAGACAAUUCACAACAAUUGAAAUUGGCUUAACCCUACUGCACUAACCGAACUAGACAACGGCGCAGUAGGGGUAAAUUCUCCGCAUUCGGUGCGGGAG
GACUCGCCGAGGCCACGCCGAGUAGGACCGAGGGUACAGCGAGUUGAUUUACCCCAUUAAUUUGGGAUGCUAAAGUCAUUUAAUGCUGACCUCCACUGGGUGGAUUAAGG
UCAAGGUAUGAAGUCCUAUUCGCUCCUGAUAGGAUCGACUUCAUAUUGCUUAUACUAUGUAACGCACAUAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGCCCCUAAGGG
AUGCAGUAGGACUAGCAUAAUGAGGGGGGUAGCAACAGUGGUGAGUUCGUUGGAUGGCUUAAGCCCUGAGUACAGGGUAGUCGUCAGUGGUUCGACGCCUUAGUAUAAAG
GUCUCGAGAUGCCACGUGGACGAGGGCACGCCCAAAGCACAUCUUAACCUGAGCGGGGGUCGCUCAGGCGAAAGCAGUUUAACCAACUGUUACGAAUACAGCCUGAUAGG
GUGCUGCAGAGGCCCACUGUACUGCUACUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUUCACAGGACGCUAGUUGUGAACAAGGUGCGAAGAGCCUACUGAGCUACUA
GAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCCAACCACGGAGCGGGUGCUCACAAGCCAGUGAGUUGCCUGUCGUAAGGCGUAAGUCUGUGGCGGAACCGACUACUUU
GGGUGUCCGUGUUUCUAUUUAUCUCAUAUAUGGCUGCUUAUGGUGACAAUCAUUGAUUGUUAUCAUAAAGCGAGUUGGAUUGGCCAUCCGGUGCCCCAGUCCCUAGGUCA
AGUGAAUGACCCCGUGAUGUUGUGAGGACAAGCGACUGGUGAGUACGCUAUGCGCGCAAGGAAGGACAUGGCUGUCCUUGGGUACUAACUACACCCCGCCCCCAAGCGGC
CAUCCUCACCACACAUGCGGAAGAACGAGUGCUGGAACGGGGAGACUUGGACUCGGCUUGUUGAGUGCACACAACAAGAGGCGAGGCUCAGGUCGCCAGUGCGACACCCGUU
CCCACACAACGGGCAUUGGGGUGCAACUCCCCCGUCCAUCCUGGACGUCACCAGGACCAGGAGAACUGUUAUCUCGGGGAGAUCCCGAUCUCUCCUACCAGCAACUCGAG
AUCUCUACGAGAUUACAGUUUUUUAAAACAGCCUGUGGGUUGUUCCCACCCACAGGGCCCACUGGGCGCAAGCACACUGGUAUUACGGUACCUUUGUGCGCCUGUUUUAC
AUCACGGCCCAACUUGGCUGAAGCUGUAAGCCAAGGGAAGGACUAGAGGUUAGAGGAGACCCCGUGCCAAAAAUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGC
UGGGCAACAUUCCGAGGGGACCGUCCCCUCGGUAAUGGCGAAUGGGACGCAAACCAUCCUUGUAUCCCCCCUCCCUGUAGAGCAAAGUGGUCUCCGGAACCGGUGAGUAC
ACCGGAAUUGCCAGGACGACCGGGUCCUUUCUUGGAUCAAACCCGCUCAAUGCCUGGAGAUUUGGGCGUGCCCCCGCGAGACUGCUAGCCGAGUAGUGUUGGGUCGCGAA
AGGCCUUGUGGUACUGCCUGAUAGGGUGCUUGCGAGUGCCCCGGGAGGUCUCGUAAACCGUGCCCGGCUUAGGUGAUGAAUUAGAAGAAGUCAUCGUUGAGAAAACGAAA
CAGACGGUGGCCUCAAUCUCAUCUGGUCCAAAACACACACAGGCUCGCGCCCGUAGUCUGGAGAAGCAUCGCCAGGGUUGAGUCGCGGCAGAACCCGGUUCGAGGACGGC
CGCGGCGAGCGGGACUUGGUCACCCCGCCGAUAUAAAGACCCACAGCCAGCCGACUUCUCCAGUUACGGGAGCGAGCCCCC
>0.302-0
AGUAGGACUAGCAAAAUAAGGGGGGUAGCAACAGUGGCGAGUUCGUUGGAUGGCUGAAGCCCUGAGUACAGGGUAGUCGUCAGUGGUUCGACGCUUUGGAGGACAAGCCU
CGAGAUGCCACGUGGACGAGGGCAUGCCCACAGCACAUCUUAACCUGGACAGGGGUCGUUCAGGUGAAAACGGUUUAACCAACCGCUACGAAUACAGUCUGAUAGGAUGC
UGCAGAGGCCCACUGUAUUGCUACUGAAAAUCUCUGCUGUACAUGGCACAUGGAGUUAUGCAGGUUUCCCCAACUGACACAAACCGUGCAAUCUGAAACUCCGCCUGGUC
UUUCCAGGUCUAGAGGGGUGACACUUUGUACUGUGCAUGACUCCACGUUCGAUCCACUGGCGAGUGCUAGUAGUAGUACUGCUGUUUCGUAGCGGAACACGACGGCCGUG
GGACUCCCCCCUUGGUAACAAGGACCCGCGGGGCCAAAAGCCACGUCUCUGACCCGUCGUGUGUGCAAACCCAGCACAGCAGCUUUACUGCGAAAACCACUUUAAGGUGA
CAUUGAUACUGGUACUCACGCACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGCAACAAGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUGUAAAA
GCGCCCAGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUCCCUCUACUACCACUGACUCCUGGGAAUAGACUGGGAGAUCUUCUGCUCUAUCU
CAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUGGCUGGUGGUGAGGAAGAACACAGGAUCUUGAUGUACCCCAUUAUUUUGGGAUGCUAAAGUCAUUUAAUGCUG
ACCUCCACUGGGUGGAUUAAGGUCAAGGUAUGAAGUCCUAUUCGCUCCUGAUAGGAACGACUUCAUAUUGCUUAUAUGUGUGCUGAUGCACACAUAUAAAUGCUCAUGCG
AAACUGCAUGAAUGCCCCUAAGGUGACUAGCGGAUGCUAGAAGAAACCCCAGUUCCUAGAUCUAGGAAAGAUCCCGUGGCGUUGUGAGGACGGGCCAGGGGGAAAGAAAU
GCUACAUGAUGAAACACCUAAAAUGGGCAAGCAGGGAGCUGGAAAGAUUUUCACUUAACCCUGACCGGUCUCCGAGGAGGUCGGGGUUGCCCACCGUUUGAUCUACAACG
GUGCGAAGCUCCGUACUUACACGCGGGCUGUGGCCGGCAUGGCCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACGAUCCGAGGGAGCUACUCCUCUCGAGAAUCGGCAA
AUGGGGCCCCCCAACGUCAGGCCACAAAUUUGUGCCACUCCGCUGGGGAGUGCGGCCUGCGCAGCCCCAGGGGGACUGGGUAGUUGUCAGACCACACGCAAGUGUGCUAC
UCUGCGAAGAGUACUGUCUGCGUUAGAGCCCCAGGAGGACUGGGAGAUGAGUCGUGCUAAGCACACUGAUGAGUCUCUGAAAUGAGACGAAACUCAUAAGUGGCUAUCCC
UCCACUUAAAUCGAAGGGUAUGGGCGGCGCAAGAGAGAAGCCCAAACCAAUUACCUACCCAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGACAGCCCAUUCCUCAG
AGCUUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACUGAUAAUGACCAUGCUAAUGCCAGAGCGUUUUCGCAUCUGGCUUCAAAACUGAUCGAAA
CGGAGGUGGACCCAUCCGACACGAUCCUUGACAUUGGAAGUGCGCCCGCCCGCAGAAUACACCACUAGCACAAAUCGGAUCCUGGGAAACAGGCAGAACUACGGUUCGUA
AGCUCGGGUAGGCCGUCAACCUACCGCCGUAUCGUAUUGUGUUUGGCCAAGUUACUGUCCCAGGGGGGGAGGGCCGCCACUUGUGGCAAGUACCUCUUCAACUGGGCAGU
AAAGGGAAUUUAAUUCGCUUCGAUUCAAGAAUGAGAUACGAAGCAAGAAUAGAAAGCGCUUAUGUGACCGUCGUUGGGAUGGAACGCAAUUAAAUACAUGUGUGACGUGU
AUUUGCGAACGACGUAAUUAUUUUUCAAUGUUAUCUUCAGCCGAACAGACAGCCUAUGCUCAAGCCAUUACUGGAGCCCACAAGGUAAUCUAGGUGGUAAUUGCGCGGGU
AAUGUCAUUGGUAAUGAUGCUCUACAACAGCGAAGCGGAACGGCGAAACACACCUUGUGUGGGUAUAUUACCCGUUGGUAGGGACAGUAUUAGUAGGACCAACACCUGUAA
AUAUAAUGGAAGAAAUAUUUUGACAAAUUGGGUUGUACUCUAGUGUUCCCAAUUAGUUCAAUUGAAACUGUCCCAGUGCUGGCGGGGGAGACAUUUAUCACAGCAUGUCU
CAUGCCCGACCCCGCUAUAGUGGCCAUCCGUAUCACAUGCGGAAAAACGACUGCUGGAACGGGGAAUCCUUGGAGAUGGAUGCUGGAUCCUGGAAAACAGGUUUAACGGG
UAGGGGGUGGUGCCCCUACCGACGCAUCACUCCAGAUACCAAAUAUGACCGGUGUCAUAGCCUCUAGCACACAGAGGUCGCGUGGGUUCAACUCCCCCCCUUUUCCGAGG
GUAUCGGAAACCAGCGAGUAAGACAGACUCUUCUGUCUGAUUGAGUGGAAGCAAGUACUAACCUAGAUUGCAUGUGGGUUAGCAGGACGCUCUGAUACAGACAUGGUGUG
AAGAGCCUAUUGAGCUAGUUGGUAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACUGCGGAGCACACGUUCGCAAGCCAGCGAGUGGUGUGUCGUAACGGGCAACUCU
GCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCUUUUAUUCUUACCUUGGCUGCUUAUGGUGACAAUUGAAAGAUUGUUACCAUAUAGCUAUUGGAUUGGCCGUCCA
GUGAAUUAUGGUGGGCUGGGCAGGGGAGCCGAAAGGCUCGCCGGUUUCCUUGAGCGCCGGUAGUUCCAACCCUGUUCAGUCCGCCACCAGCCACUGGCAUGCAGCCGUUA

UGAUGACGAGUCAGGUCGGGCUCCGCGGUGGGUUUGGUCGCCCAUGGGACGGAGAUAUGGAAAGGGUCUCGUACGCUGUAAGACUAACUUCGGUUGAGCUUGCGAUAUGG
GCCUAUGCCUGGAUAAGUCAUAAUGUUACUGCCCAGUGAGACGUUCAUCUGCGUGAGCUUCCGAUCUCAGUAUUGUUUGGAAGGAUCAUUUGAUUAACGCGGUUUGAACA
GUUUUUUGGAGCUUUUGAUUUCAAAUGUCUAAAAAACCAGGAGGACCCGGGAAGCCCCGGGUCGUCAAUAUGCUAAAACGGACUCGGCUUGCUGAGUGCACGCAGCAAGA
GGCGAGGCGGGGAAAUGGACUGAGCGGCGCCGACCGCCAAACAACCGGCAGGUCUCUCUUGUUAGACCAGAUCCGAGCCUAGGAACUCUCUGGCUAACUAAGGAACCCAC
UUAAAACAGCCUGGGGGUUGUACCCACCCCUGGGGCCCACGUGGCGCUAGUACUCUGGUUCGUUAGAACCUUUGUACGCCUGUUUUCCCCGCCCGCGCAAGGAAGGACAC
GUAAAUCACGUGUUCUUGGGAGUUGACGACUCUCCGCCCCCAAUGAAUAACCAACGGAAAAAGGCGAAAACCACGCCUUUCAAUAUGUUUCAUCGUGGCCACGCGGAGUA
CGAUCGAGGGUACAGUGAAUUAGGAGAGUCAGGGCCAGGACCCCCCCCCUGAGCCCAGGAUAACCUUAAAAAUAGGGGGGCAAAUAAACAACUACAUACAGUUCAAGAGC
AAACACCGUAUUGAACCAGUAUGUCUGUUGGUAUUUUUUAGAGAAAAUUUAGCCUUCCAACAAGGGGAGGCCAAAGAAUUUCCUUUCAGCUGAUCGUUGUUGUCAUUCUU
CAAUUUUUGCAGGUAACUCAGAAAGCGUUUCUUGAGCUUCCCGGGGAUGAGCGUCUGGGAGAGCUGAUUCUGCAGACCGGCUAGGAGCCGCAGGGCGUUCUCCGGGGCGC
GGCCCGAGACCGUCGCGUCGUAGACAGCCAGGAGACCCGGGAGCAGAUUCACCGCAGCGGCCGUCUCCCUCAAGGUGCUGUGAAAUUCUCCAGGAUUUCUCGACCUCGGU
UGUUUUAUCUGGUAGUUUGCGCUAAUCUUCGCUUUGCGUAGGGAAUUUCUCUCCUUAAUACAUUUCUGAGUUAAGGGGUUUGAAGGUUGGGUCUAUCUUUUGAGUUUAUC
UCACUAGAGGAUAAAUUGUAACUUCCAACAAAUAAGCGAUAUGAGGGGGCCCGAAGCCAGGAGGAAGCUGUACUCCUGGUGGAAGGACUAGAGGUUAGAGGAGACCCCCC
CAACACACUUUGCCUACUCCUACUUAGCGUAGGAGUAGGCAUCUGGUGCUUUUAUGUUCACCGAAAUCGAACAUACUCGGCCGUAAACACCCAAUCAUCAGUAGGCCGAG
CCCGCCCCCUGAUGGGGGCGACACUCCGCCAUGAAUCACUCCCCUGUGAGGAACUACUGUCUUCACGCAGAAAGCGUCUAGCCAUGGCGUUAGUAUGAGUGUCGUACAGC
CUCCAGGCCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUUGCCGGAAAGACUGGGUCCUUUCUUGGAUAAACCCACUCUAUGUCCG
GUCAUUUGGGCGUGCCCCCGCAAGACUGCUAGCCGAGUAGCGUUGGGUUGCGAAAGGCCUUGUGGUACUGCCUGAUAGGGUGCUUGCGAGUGCCCCGGGAGGCCUCGUAG
ACCGUGCACCAUGAGCACAAAUCCUAAACCUCAGGCUCGUCUGCCGUAGUCUGGAGAAGAAUCGCCAGGGUUGCGUUGCGGUGUGCCCCGGUUCGAGGCCGGCCGGAUUC
CGCGGCUAACGAGGGCGUGGCUGCCCCGUCGUUUCCAAGACCCCAUAGCCAGCCGACUUCUCCAGUUACGGAGCGAGCCCCUAUGAGCCCCCGGGGAAAGAAGCCCCGCA
AUUUCCCCGUGGCUCAAGGUCUGGGUGAUGAACUGGAAGAAGUUAUUGUUGAGAAAACGAAACAGACACUAGCUUCAAUAUCAUCAGGACCUAAGCAUACCCAGCCUCCA
GGCCCUGACUGACCUGGUUUCCAGGGCCCUGGAGGCGACCACGUUCCUUUAAACUGCCAGGGGGCGGGCUCGAGCCUUCGGGCGAGUCCGCCCUUGGACGGCUUAUCCAA
CUCGGAGAAAGACCUACGCUGCCCUAGCGGUUCUGCUAGGGAGGAGAUUGGUGCGGCUUAUGCCGCCCGUCCCGGGUACAAGUCCCGGGAGGUGACGGUAGCGUCUGUAU
UUGCCCUGUAUCAGGUGUAUAGCUGCUCUCUCGCAGUCGCUGCUAUCAGCCGUCUCCGGUCCGAUGCCAGCUGGUGGUCUGCUCUUUU
>0.302-1
AGUAGGACUAGCAAACGGAGGGACUAGCCGUAGUGGCGAGCUCCCUGGGUGGUCUAAGUCCUGAGUACAGGACAGUCGUCAGUAGUUCGACGUGAGCACUAGCCCACCUC
GAGAUGCUACGUGGACGAGGGCAUGCCCAAGACACACCUUAACCCUGGCGGGGGUCGCUAGGGUGAAAUCACAUUAUGUGAUGGGGGUACGACCUGAUAGGGUGCUGCAG
AGGCCCACUAGCAGGCUAGUAUAAAAAUCUCUGCUGUACAUGGCACAUGGAGUUACGCAGGAAUCCUAGACCAACCCAACCCGUGCAACUGCAAGUUUCGCCCGGUCUUU
CCGGGUCUAGAGAGACGAACAGAUGUACUGAGAUCAACUCCACGAUUGGUCUACUAGCGGGUGUUAGUAACACUCACUUUGCUUCGUAGCGGAGCGCAUGAGCGGCGGGA
ACUCCCCCCAUGGUAACAUGGACCCUCCGGGCCAAAAGCCACGCCUCACGGCCUCAUGCGUGUGCAACCCCAGCACGGCAAUUUGUUUGUGAAACACACCUUAAGGUAACA
CUGAGACUGGUACUUGGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGAGACACUCGGGAUCUGAGAAGGGGACCGGGAGUUCUAUCAAACU
GCCCGGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCCUAUUUAAACCUUGAUUUCUGGGAAAGACCAGAGAUCCUGCUGUCUCUGCAACA
UCAAUCCAGGCACAGAGCGCCGCAAGAUCGAAUGGUGCUGUUGAAUCCAAGGGCACGCCGAGCUUGAUGUUUUCUUGACAUAAGUCAAAUUGCCAACCUCCACUGGGUGG
GUCAAGGUUGAGGUAUAGAAUCCUAUUCGCUCCUGAUAGGAGAAAUUCUAUAUUGCUUAUAUAUGUGCUUACGCACAUAUAUAAAUGCUCAUGCAAAACUGCAUGAAUGC
CCCUAAGGGAUGCUGACUAGCGGAGGCUAGGGGGGGCCCCAGUCCGAUAGGUCAUGGAAUGACCCCAUGGUGCUGAGAGGGCAGGCCAGGAAGAAAGAAAAAAUAUAAAU
UAAAACAUCUAGUAUAGGCAAGCAAAGAGCUGGACAGAUUUGCACUUAACCCUAGCCAGUUUGUGAGAAGGUUGGGGUGGCCCACCGAUAUGGGUAUGAUGGUUUCCUGU
CAUACUACAGUGGUGCGAAACUCCGUACUUACACACGAGCUAUGGCCGGCAUGGUCCCAGCCUCCUUCGUGGCGCCGGCUGGGCAACAUUCCGAGGGGACCGUCCCUCGG
UAAUGGCGAAUGGGACCCACCCGUGUCAGGUCACAUUGUGCCACCUUGCCAGAAGGUGCAACCUGCGAGGCCCCAGGUGGACUGGGUCAACGUCAGGCCACAAUUCUGUG
CCACUCCGCUGGGGAGUGCGGCCUGCGUAGCCCCAGGAGGACUGGGUGAAAGGUCUGUGCUUAGCACACUGACGAGUUCCUGAAAUGGAACGAAACCUUUUAGUGUUUUU
CUGUCCACUUAAAUCGAACAGAAUUGGCGGCGUAGUACACACUAUAAAGAAUCAAGCCGACCAAUUGCACUACCAUCAUCAUGGAGAAACCUGUAGUUAACGUUGACGUA
GACCCUCAGAGUCCGUUCAUUACGCAACUGCAGAAGAGUUUCCCGCAGUUUGAGGUAGUAGCACAGCAGGCCACACCAAAUGACCAUGCAAAUGCCAGAGCUUUUUCGCA
UCUGGCCAGCAAACUGAUCGAGCUGGAGGUCCCGACCACAGCGACGAUCUUGGACAUCGGCAGCGCACCUGCUCGUAGAAUGCGCCACUAGUGCAUCGAAUCCUGGGAAA
CAGGCACGGAGAACAAAAUCCGGUAAGCUCGGGAAGCCGUCAACUUCCGCCGUAUUGAUGAAAACAACUCCUUAAUUGUCCAAGGAGGAAGAGCGGCCAUAUGUGGCCGC
UACCUCUUCAACUGGGCAGUUAAGGGAAUCUAAUUCGUUAUCUGUUUGAAGGUGGGUAUGGCAGAGUAAGAAUAGAAAGCGCUUAUGUGACCGUCGUCUGGAUGAAACAC
AAGAUUUACACGUGUGACGUGUAUAUGUGGGCGACGUAAAAUUUUUACUAUGUCAUCUUCAGCCAAUUCGACAGCCUGAGAGUCAGCUCUAACCAGAGCCCACAAGGUAA
UCUGGGGAGUAAUGAACCCGGUAAUGUCGGUGGUAAUGAUGCUCUGAAACAGAGAAGUCAACCAGAGAAACACACGUUGUGGUAUAUUACCUGGUAAUAGGUACAGUAUG
AGUACGACCUACACCUGUCAACAUAGUGAACGAAAUCUGUAGACUGAAUGGUUUGCACUCUAAAUUUUCCCAUCAGUCCUAUUGAAACUGUACCAGUUACAGCGGGGGGG
ACAUAUUUCACAGCGUGUCCCGUGCCCGACCCCGCGUAAGUGGGUACCCCCACCACGUGCGCAAAAACGAGCACUGGAAUGGGGAAUCUUGAGGUAUCGAUGCUGGAUCC
UGGGAAACAGGCUUGACAGGUUUGGGUGGUUCCAAACCGCUGCAUCACUCGAGAUACUCAAUGAAGGCAGUCCUUCUCCCUUUAGCACACAAAGGUCAAAUGGGUGCGAC
UCCCCCCCUUUUCCGAGGGUCAUCGGAAACCAGCGAGUAAGACAGACUCUUCUGCCUGAGUUCGCGGAUACAAGUGUGAAUCUAACAACGCAUACAGGUUACACAGGACG
CUAGAUUUGAACAAGGUGCGAAGAGUCUAUUGAGCUACCAAAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCCAACCACGGAGCAGGCGCUCACAAUCCAGUGAAUAGC
CUGUCGUAACGCGUAAGUCUGUGGCGGAACCGACUACUUUGGGUGUCCGUGUUUCCAUUUUCUCUCUUGUAUGGCUGCUUAUGGUGACAAUCAUUGAUUGUUAUCAUAAA
GCGAACUGGAUUGGCCACCCGGAGUAUUAUGGUGGGCUGUGUGGGGGCACCGAAAGGUGCGCCGGGUCCUUUUAGCCGGUUACGCCAACCCUGCACAGUUCACCACCAAC
UAGUGAGACUAGGUUCGUGUGAUGACGAGUCAGGUCGGGCUCCGCGCUGGGUUUGGUCACCCAGGGGAUGGAGAUAUGGAAAGGAUCUCGUGCGCUAUAAGUCUAACGAA
AGUUGAGCUUGCAACAUGGGUCUAUGCCUGAAUAAGUCACGUGAGUUGCCACCACUAAGUCGUUCGUCUGUGUGAGCUCUACUACUUAGUAUUGUUUUUGGAGGAUCGUG
AGAUUAACACAGUGCCGGCAGUUUCUUUGAGCGUUGAUUUUCAAUGUCUAAGAAACCAGGAGGGCCCGGAAGAAGCCGGGCCAUCAAUAUGCUGAAACGGACCCGGCUUG
CCUGAAGCGCGCACGGCAAGAGGCGGGGGGGGGAAACGGGAGAGCGGCAGCGCCCGCCGAAACAACCGCAGGGUCUCUUUCACUGGACCAGAUUUGAGCCUGGGAGCUCU
CUACCAGUGAAGGAACCCACUUAAAACAGCCUGUGGGUUGCACCCACCCACAGGGCCCAAUGGGCGUCAGCACUCUGGUACCGAGGUACCUUUGUGCGCCUGUUUUAUUU
AUGCGCGCAAGGAAGGACAUGGCUGUCCUUGGGUACUAACUACACCCCGCCCCCAAUGAACCAACGAAAAAGGGUGGUUAGACCACCUUUCAAUAUGCAGUGCCGGGGCC
ACGCGGAGUACGAUCGAGGGUACAGCACUUUAGGGGAGACGGGGUUCGGACCCCCUCCCCGAGCCCAGGGUAACUUUGAAGGUGGAGGGGCAGAUGAGUAAUUACAUACA
GUUCAAGUCCAAAUGCCGUAUUGAGCCUGUAUGUCUGCUCCUGUUUUUUAGGGAGAGCUUGGCCUUCCCUCAAGGGAAGGCCGGGGAACUUCCUCGUGGUGUAAAGUUUU
GCCUGAACCUGUGGUUGGGCAGGUAACUUAGGAAGCGUUUCUUGAGCUUCCCUGGGAUGAGCGUUUGGGAGAGCUGAUUCUGCAGCCCAGAGAGUAGUCUCAGGGCAUCC
UCUGGAGCCUGACCUGUGAUCGUCGCAUCAUAGACCGCCAGUAGACCUGGGAGCAGAUUCACCGCCGCGGCCGUCUCCUUUAAGUGUGAAUCGAAAUAUCUCAGGAUUUG
ACCGUCCGGUGAGUUGCGCUUAUCACUUGCGUAAGGUAUUUCUCUCUUCAUUGUUCAUAUGGAGGGUUUGAAGGCUGAGUCUACCACAACUGUGGGGAUAAAUUGUAACU

UUCAGCAAACAGUGAUGAGUGGCCCAAGCUCGCCGAAGCUGUAAGGCGGGUGGACGGACUAGAGGUUAGAGGAGACCCCACUCUCAAAUGGUGCCUACUCCUACUUUCAG
UGGGGGUAGGCAUUUGGUGCUUGUUAUUUCACCUAAAUCGAAAUAACUCGGCCACAAACACGCAGUCUACUGUUGGCCGAAUCCCUGUGAGGAACUUCUGUCUUCACGCG
GAAAGCGCCUAGCCAUGGCGUUAGUACGAGUGUCGUGCAGCCUCCAGGACCCCCCCCUCCCGGGAGAGCCAUAGUGGUCUGCGGAACCGGUGAGUACACCGGAAUCGCUG
GGGUGACCGGGUCCUUUCUUGGAAUAACCCGCUCAAUACCCAGAAAUUUGGGCGUGCCCCCGCGAGAUCACUAGCCGAGUAGUGUUGGGUGGCUCGCGCCCGUAGUCUGG
ACAAGCAUCACUAGGGUUAAGUUACAGCAGAACCCGGUUCGCGGACGGCCGCGGCAAGCGGGACUUAGUUACCCCGCCAAUUUAAAGACCCGCAGCCAGCCGACUUUUCC
AGUUACGGGAGCGAGCCCCCUUUGGACCUUGGGGAAAGAAGCCUCGCAACUUCCCCAUGGCCCAGGGGUUGGGAGAUGAACUAGAGGAAGUCAUAGUCGAAAAAACGAAA
CAGACGUUGGCUUCAGUUGCAACAGGUUCAAAAUAUACCCAGCCUCCAGGCCCUGCAACACCUGGUCCGGAAGGCCUUGGAGGCGAAGGUGAGCCUUUAAACUACCAGCG
GGCGGACCUCCUACGGGAGAGUCCGCCCUUGGAAGGCUUAUUCAGCUUGGGAAGGGACCUACGCUGCCCUAGCGGUUAUGCUGGGGAGGAGAUGAGUGUGACUUAAAUCA
CCCGUCCCGGGUACAAGUCCCGGGUGGUGAAGACGGGGUCUGGUGCUGCUAAAAGUUUGGACAGUCCCCGGUUUCUGCUCUCUUCGGUUCAAACCAGCUGGUGGUCCGCC
UGUUU
>0.351-0
UUUUUUAGGGAAGAGCUGGUCUCCCUUCAAGGAGACCAGGAAACUUCCCCGAAGAGUCUGCGCUAAGCGCACUGAUGAGUCUUUGAAAUAAGACGAAACUCUUCGCAAUC
CGGCCUUUAAACUACCAGCGGGCGGGCUCCCUACGGGGCAGUCCGCCCUUGGCCAACUUAUCGAGUACGCGGGAGUGUUUGCCGUUCCACUUAAAUCGAAACGGUUAGCU
CGCCAGUUAGCGAGGUCUGUCCCCACACGACAGAUAAUCGGGUGCAACUCCCGCCCCUUUUCCGAGGGUCAUCGGAACCACUGGGAGAGACCAGAGAUCCUGCUGUCUCU
ACAGCAUCAAUCCAGGCACAGAGCACCAGAAAAUGGAAUGGUGCUGUUGAAUCAACAGGUUCUGUAAGUGGACUUUAAUUUUUUCUGCUAAGCCCAACACUCCACCACAC
CCAGGCACACACUACACACACCCACCCGUCUCAGGAAAUUCCUCAGAUUUCUCCCUGUUGACCAGGAGUUAUCGGUGAGUUGCGCUUUGUGCUUUGCGUAAGGGAUUUCU
CUCGUACAGGUACGAUCUCGGACCAGCUACGGGUUUGAAGGGUGAGCCUAUUCUUCGGAAGGGUAAAUUGUAACUUUCACCAAUAAGCAAUGCGACCUACUGCGGCAGCA
CACACAGUGACGGGAAGUUGGUCGCUCCCGACGCAGUUAGGUCAGAGAAAAUUUUGUGAGACCAAGGCUCGCUCCCUGUAGCCGGAGGGUUAUUUUCCAAGGGUUGAGUC
GCAGGACCCCCGGUUCGAGUCUCGGGCCGGCCGGACUGCGGCGAACGGGGGUUUGCCUCCCCCGUCAUGCAAGACCCCGCUUGCAAAUUCCUCCGGAAACAGGGACGAGCC
CCUGCGCCACUAGUGCAUCGAAUCCUGGGAAACAGGCACGGAGAACAAAAUCCGGUAAGCUCGGGAAGCCGUCAACUUCCGCCGUAUUGAUGAAAACAACUAUGGGACGC
UACAAUACUGACAUGGUGCGAAGAGUCUAUUGAGCUAGUUGGUAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACUGUGGAGCAGAUACCCACGCACCAGUGGGCAGU
CUGUCGUAAUGGGCAACUCCGCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCUUUUUAUUCCUACGCUGGCUACUUAUGGUGACAAUUGAGAGACUGUAACCAUAUU
GCUAUCGGGUUGGCCACCUGGUGCCCCAGUCCCCCAGGUUAUGGGAGAAUAACCCCGACGUGCUGGAAGGGCAUUAAAACAGCUCUGGGGUUGUUCCCACCCCAGAGGCC
CACGUGGCGGCCAGUACACCGGUAUUACGGUACCCUUGUACGCCUGUUUUAUACAGCGAGUAAGACAGACUCUUCUGUCUGAUUGAGUGGAAGCAAGUACUAACCUAGAU
UGCAUGUGGGUUAUUCCUGGAGCCGGUGACCUGAGACCCACCCUUCUCGACGGGGCAGGUGUUAUAAGUGUCCUGUAGAGUAUGGGUGUGGUUUUAGUGCACGCGGGAGA
UUGGCGGCGUAGUACACACUAUUGAAUCAAACAGCCGACCAAUUGCACUACCAUCACAAUGGAGAAGCCAGUAGUUAACGUAGACGUAGACCCUCAGAGUCCGUUUGUCG
UGCAACUGCAAAAGAGCUUCCCGCAAUUUGAGGUAGUAGCACAGCAGGCCACUCCAAAUGACCAUGCUAAUGCCAGAGCAUUUUCGCAUCUGGCCAGUAAACUAAUCGAG
CUGGAGGUUCCUACCACAGCGACGAUUUUGGACAUAGGCAGCGCACCGGCUCGUAGAAUUGUUGGGUUCUAGAUGUUAUGAUGACGAGUCGGUCGGCUCCGCGCUAGGUU
UGGUCGCCUAGGGGAUGGAGAUAUGGAAAGGGUCUCGUGUGCUGUCAGUCGGUCGAAAGACGCGCUUGCAACAUGGGCCUAUAACCGGAUAAGUCAUAGCAAUACUAGCC
AACAUUUCAUCGUGGCCACGCGGAGUACGAUCGAGGGUACAGUGAAUGGGUCUCUCUAGGUAGACCAGGUCCAAGCCCGGGAGCUCCUGGCUAUCUAGGAAACCCACCCA
CCGUCAGACCACACCGAGGUGUGCCACUCAGCGGAGAGUGCGGUCUGCGCAGCCCCAGGAGGACUGGGUGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACC
CUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAAGUCUUAGAAUGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGUCGGCCGUAAACACCCAAUCAU
CAGUAGGCCGAGGACCCGGCUUGCCUGAAGCGCGCACGGCAAGAGGCGGGGGGGCGGUAAGCAAGACGGAUCCUGGGAAACAGGUUUAACGGGCUGGGGGUGGUGCCCCG
GCCGUCGCAUCACUUGCUGAUACAACAAGGGACGUGUUGGUGGGACCAACACCUGUCAAUAUAUUGGGAGAAAUAUUUUAACACAGAUAGGGUGCACACUGGUAUUCCCU
AUCAGUUCAAUUGAAACUGUCCCAGUACACUACAUAUUUAUUUAUUUAUUUCUUAGUAAUGUUUACUGGAACUUAAAUCUGUGAUAACCUAAACUAAAAGUCUCAAACAA
CCCGUUACUUGCUGACCAAUUUUUGUAGGUACUGGGUGUAAAUAUGAUGACCGGUACCAGCCGGCGGGGGCGACAUAUAUCACAGCGUGUCGCGAGCCCGACCCCGCAUC
GUCGUUAGGAUGGAACGCAAUUAAAUACAUGUGUGACGUGUAUUUGCGAACGACGUAAUUAUUUUCAGGCCACCGCAAGGGAGAGGGAUCCCCUCUCGGGUUUGGACGAC
ACCCCGGCCCCACACCGUCAGACCACACCGAGGUGUGCCACUCAGCGGAGAGUGCGGUCUGCGCAGCCCCAGGAGGACUGGGUAUGAAUGACCAACGGAAAAAGGCGAGA
AACACGCCUUUCAAUAUGGGUAGUGGAAGAGGUCGGAACUGCCCACCGCCUUGGUUAUGACGGGUUUCUGUCAUACUAUAGCGGCGCGAAGUUGCGUACUUAUUCGAAUG
CCCAGUAGUUCGCCUGUGUGAGCUGACAAACUUAGUAGUGUUUGUGAGGAUUAACAACAAUUAACACAGUGCGAGCUGUUUCUUAGCACGAAGAUCUCGAUGUCUAAGAA
ACCAGGAGGGCCCGGCAAGAGCCGGGCUGUCAAUAUGCUAAAACAGCCAGGAUCUAAAAAGAAAUAUAGGCUAAAACAUUUAAUAUGGGCAAGCAAAGAGCUGGAAAGAU
UCGAAUGUAAUCCUGGUCAACUCAUCGCCCAAGGGGGGAAAGCGGCUAUAUGCGGUAAGUACCUCUUCAAUUGGGCGGUGAAGAGUUAUGGUGGGCUGGGCGGGGGCAUC
GAAAGAUGCGCCGGUUUCCUUGAUCACCGGUUACGCCAACCCCGUUCAGUUCACCACCAGCAUGUCAUCUUCAGCCAAUUCGACAGCCUGGGAGUCAGCUCUAACCAGAG
CCCACAAGGUAAUCUGGGGAGUAAUGGUAAACCCGGUAAUGUCGGUGGUAAUGAUGCUCUGGUUGAACAGGAUGUGGCACCCUAAGAGGACGCAGGCAUACAAGGUUAUU
ACCCAGUCCUUGUAUGCCUGGUGUCCCCUUAGUGGGACGCAGGCCUAGGUAGCAAACAGGCUCCCCCAACUGACACAAACCGUGCAACUUGAAACUCCGCCUGGUCUUUC
CAGGUCUAGAGGGGUGACACUUUGUACUGUGUUUGGCUCCGCGUUCGAUCCACCGGCGAGUGUUAGUAACACCACUGCUACUUCGUAGCGGAGCAUGACGGCCGUGGGAA
CCCCUCCUUGGUAACAAGGACCCACGGGGCCGAAAGCCACGUCCACAUGGACCCGUCAUGUGUGCAACCCCAGCACGGUAGCUUUGCUGUGAAACUCACUUUAAAGUGAC
AUUGAUACCGGUACCCAAAUACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACAUGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUGUAAAAG
CGCUCAGUUUAAAAAGCUUCUAUGUCUGGAUAGGUGACCGGAGGCCGGCACCUUUUCUUUUACUAUUUGACUCUAUAGGGAGGUCAGGGCCAGGAGCCCCCCCCUGAGCC
CAGGAUAACUCUUGAAGUUGGGGGGCAUUUUGCCUACUCCUGCUCACUGCAGGGGUAGGUAUCUUGGGGGCCCAAGGCGAGAUGAAGCUGUAGUCUCGCUGGAAGGACUA
GAGGUUAGAGGAGACCCCCCCGAAGCAUAUUAACAACUACAUACAGUUCAAGAGCAAACACCGUAUUGAACCAGUAUGCUUGCUAGUGUCAGCUGAUCGUUGUUGUCAUU
CUUCAAUUUUUGCAGGUAACUCAGAAAGCGUUUCUUGAGCUUCCCGGGGAUGAGCGUCUGGGAGAGCUGAUUCUGCAGACCGGCUAGGAGCCGCAGGGCGUUCUCCGGGG
CGCGGCCCGAGACCGUCGCGUCGUAGACAGCCAGGAGACCCGGGAGCAGAUUCACCGCAGCGGCCGUCUCCCCUCAAGGUGCUGUGGGUCUUAAUGAUGAAUUAGAAGAGG
UCAUUGUCGAGAAAACGAGACAGACAUUGGCCUCUGUUUCUUCUGGACCUAAACAUACACAUAGCGCUAACGCAAUUAGUGGAAAAAGAAUUGCAGCUAGGACAUAUAGA
ACCUUCACUUAGUUGUUGGAACACACCUGUCUUUGUGAUCCGGAAGGCUUCCGGGUCUUAUCGCUUAUUGCAUGACUUGCGCGCUGUUAACGCUAAGCUUGUUCCUUUUG
GGGCCGUCCAACAGGGGGCGCCAGUUCUCUCCGCGCUCCCGCGUGGCUGGCCCCUGAUGGUCCUAGACCUCAAGGAUUGCUUCUUCUCUAUUCCUCUUGCGGAACAAGAU
CGCGAAGCUUUUGCAUUUACAGUAGGACUAGCAAAAGUAGGGGACUAGCGGUAGCAGUGAGUUCGUUGGAUGGCCGAACCCCUGAGUACAGGGGAGUCGUCAAUGGUUCG
ACACUCCAUUAGUCGAGGAGUCUCGAGAUGCCAUGUGGACGAGGGCAUGCCCACGGCACAUCUUAACCCAUGCGGGGGUUGCAUGGGUGAAAGCGCUAUUCGUGGCGUUA
UGGACACAGCCUGAUAGGGUGUAGCAGAGACCUGCUAUUCCGCUAGUAAAAACUCUGCUGUACAUGGCACAUGGAGUU
>0.351-1
UUUUUUAGGGAAAAUCUGGCCUUCCCACUCGGAGGCCAGGGAAUUUUCUUGAAAGGUCUGUGCUUAGCACACUGACGAGUUCCUGAAAUGGAACGAAACCUUUUGAAGGU
GAGCCUUUAAACUACCAGCGGGCGGACCUCCUACGGGAGAGUCCGCCCUUGGAAGGCUUAUUCAGCUUGGGAAAGUGGCUAUCCCUCCACUUAAAUCGAAGGGUUCAGGU

CGCCAGUGCGACACCCGUUCCCACACAACGGGCAUUGGGGUGCAACUCCCCCCGUCUAUCCUGAACGUCAUCAGGACCACUGGGAAUAGACUGGGAGAUCUUCUGCUCUAU
CUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUAGCUGGUGGUGAGGAAAAACAGUAAGUGUCUUUAAUUUUUAAGCCUCAUGCACAGCACACAAACACCAGGC
ACACAAACAGCAGCCACACUUCCACACGCGCCCGUCUCCAGGAAAUUCCCCAGGAUUUCUCGACCUAGUACGUCUACCUGGUGACUUGCGCUGUCUGUUGCUUUGCGUAG
GGAAUUUCUCUCCAAUAUUAUUAUCUUUAAUUUGGGGUUUGAAGGUUGGGUCUACCCUUCGGGGGGAUAAAUUGUAACUUCCAACAAAUAAGCGACAUGACGACGAACUG
UGGCAGCACACCAUUUGGUGACGGGAAGCUGGUCGCUCCCGACGCAGGUCGUCGUAAGAACUUUGUGAGACCAAAAGACUCGACUCUGUAGCCUGGAGGAACGUGAACGG
GUUGGGUCGCGGUGUACCCCGGUUCAAGACUUGUACUCGAGCCGGCCGGAGCCGCGGCUAACGUGGUAUUGGCACUCCCGUCUCGACCCAGCCUACAAAAAUCCAGGAUA
CGGAAUCGAGUCGUUACACCACUAGCACAAUCCGGAUCCUGGGAAACAGGCAGAACCUCGGUUCAUAAGCUCGGGUAGGCCGUCAAUCUACCGCCGUAUCGUUUUGUGUU
UGGCCCACAGGACGCUAGUUGUGAACAAGGUGUGGAAGAGCCUAUUCAGCUACAUGAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACCACGGAGCAGGCGAUCGC
AAACCAGCAAUUAGCCUGUCGUAACGCGAAAGUCUGUGGCGGAACCGACUACUUUGGGUGACCGUGUUUCUCUUAUUUUUAUUCAUGGCUGCUUAUGGUGACAAUCAUAG
AUUGUUAUCAUAAAGCGAGUUGGAUUGGCCAUCCGGUACCCCAGUCCCUAGGUCAGGUGAAUGACCCCGUGAUGUUGUGAGGACAUUAAAACAGCCUGUGGGUUGCACCC
ACCCACAGGGCCCAAUGGGCGUCAGCACUCUGGUACCGAGGUACCUUUGUGCGCCUGUUUUAUUUGGCGAGUAAGACAGACUCUUCAGUCUGAGUUUGUGGAGAUGAGUG
UAAAUCUGGCAUAGCAUACAGGUUAUCCGUGUAGCUGGUGACCUAACACCCGCCCAUCACCACCGGACAGAUUCUGAACUUGUCCUGUGGUGUUUGGUGUGGUUUUGGGG
UACGCAGGAGAUGGGCGGCGCAUGAGAGAAGCCCACACCAAUUACCUACCCAAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGACAGCCCAUUCCUCAGAGCUUUGC
AACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACUGAUAAUGACCAUGCUAAUGCCAGAGCGUUUUCGCAUCUGGCUUCAAAACUGAUCGAAACGGAGGUG
GACCCAUCCGACACGAUCCUUGACAUUGGAAGUGCGCCCGCCCGCAGAAUUAGUGAGACUAGGUUCGUGUGAUGACGAGUCAGGUCGGGCUCCGCGCUGGGUUUGGUCAC
CCAGGGGAUGGAGAUAUGGAAAGGAUCUCGUGCGCUAUAAGUCUAACGAAAGUUGAGCUUGCAACAUGGGUCUAUGCCUGAAUAAGUCACGUGAGUUGCCACCACUACAG
UGCCGAGGCCACGCGGAGUACGAUCGAGGGUACAGCACUAGGGUCUCUUUCACUGGACCAGAUUUGAGCCUGGGAGCUCUCUACCAGUGAAGGAACCCACCCCGUGUCAG
GCCGCAAAGCGCCACUUCGCCAAGGAGUGCAGCCUGUAUGGCCCCAGGAGGACUGGGUGUUGCCUUUGAGAGUUACUCCUUGCUCUCUUCAGAAGUUCUCUACCGAGAAC
GUGCAGUGGCUGUCAUAGACAGCCUUCAGACUGCGGGCAACGUCAGUUCCAUAAGGAACUGAUAAUCUCCAAAAGGAGUCGGCCACAAACACACAGUCUACUGUUGGCCG
GGGACUCGGCUCGUUGAGUGCACACAGCGAGAGGCGAGGCGAGGUAUCGAUGCUGGAUCCUGGGAAACAGGCUUGACAGGUUUGGGUGGUUCCAAACCGCUGCAUCACUC
GAGAUACUCAUUGGUACGUACUAGUAGGACCUACACCUGUCAACAUAUUGGAAGAGAUCUGUUGACUCAGCUUGGGUGUACACUAAAUUUUCCAAUUAGCCCCAUUAAAA
CUGUACCAGUACACUACAUAUUUAUUGUAUUUUAUUUAUUAAUGCUGCUUUAAACCUCACACACAAGGAUUUGCUUCUACUAAUUUUUGAAAGCACUGGGUGUAUAUAGC
UAAAAACCAGUGCUAUACAACGGGGGAGACAUUUAUCACAGCGUGUCUCGUGUCCGGCCCCGUUGGGUCGUCUGGAUGAAACACAAGAUUUACACGUGUGACGUGUAUAU
GUGGGCGACGUAAAAUUUUUACUGCCCACGCAAGGGGAGAGAAGAUUUUCGCCUCCGGGUGUUGACGACACCCCGACCCCACCGUGUCAGGUCACAUUGUGCCACCUUGC
CAGAAGGUGCAACCUGCGAGGCCCCAGGUGGACUGGGUAUGAACAACCAACGAAAAAAGACGGCUCGACCGUCUUUCAAUAUGGGUUUGUGAGAAGGUUGGGGUAGCCCA
CCGACUUGGGUAUGAUGGGUUUCUGUCAUACUACGACGGUGCGAAACUCCGUACUUACACACGAGCCAGACGUUCAUCUGCGUGAGCUUCCGAUCUCAGUAUUGUUUGGA
AGGAUCAUUUGAUUAACGCGGUUUGAACAGUUUUUUGGAGCUUUUGAUUUCAAAUGUCUAAAAAACCAGGAGGACCCGGGAAGCCCCGGGUCGUCAAUAUGCUAAAACGG
CCAGGGGGAAGAAAAUUUAGAAUGAAACAUUUAGUAUGGGCAAGCAGGGAGCUGGACAGAUUUGCACUCAACCCUGACCAAGCUACUGUCCCAAGGGGGGAGGGCUGCCA
CCUGUGGCAAAUACCUCUUCAAUUGGGCUGUAAGGAUCAAUGGUGGGCUGGAUGGGGGAGCCGAAAGGCUCGCUGGGUCCUUGUUAGCCGGUAGUCUCAACCCUGUUCAG
UUCACCGCCAAUAUGUUAUCUUCAGCCGUACAGGGAGCCUUGAACCGAGCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAU
GAAGCUCUACUUACGCGGCCUGCGGAACCCUAAGAGGGGGCCUGCACGCCGAGGUUAUUUCCCAGCCCUUCGAUGCAUGGUCCCCCCUUAGUGGACUGCAGGCCCAUGU
AGCACGCAGGAAUCUUAGACCAACCCGCCCGUGCAACUGCAAGUUUCGCCCGGUCUUUCCGGGUCUUGAGAGACAAACAGAUGUACUGAGGCCAACUCCACGAUUGGUCU
ACUAGCGGGUACUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGCGGGAACACCCCCAUGGUAACAUGGACCCGUCGGGCCAAAAGCCACGCCUCACGGCC
UCAUGUGUGUGCAACCCCAGCACGGCAACUUGUUUGUGAACCACACCUUAAGGUAACACUGAGACUGGUACUUGGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUAC
CCCGAGGUAACACGAGACACUCGGGAUCUGAGAAGGGGACCAGAAGUUCUAUCAAACUGUCUGGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCU
UUUCCUUUACUUAAACCCAAUUUAGGGGAGACGGGGUUCGGACCCCCUCCCCGAGCCCAGGGUAACUUUGAAGGUGGAGGGGCAUGGUGCCUACUCCUACUUUCUGUGGG
AGUAGGCAUUUCCACGGCCCAAGCCUCGUCUAAGAUGCAAUAGAUUAGGUGUAAGGACUAGAGGUUAGAGGAGACCCCGUGGAAACAGAUGAGCAAUUACAUACAGUUCA
AGUCCAAAUGCCGUAUUGAACCUGUCUGUCUCUUGCUCAGAGGAUGAAGACUAAGUCACAGGCUUAGCCAGGUAACUUAGGAAGCGUUUCUUGAGCUUCCCUGGGAGAGC
UGAUUCUGCAGCCCAGAGAGUAGUCUCAGGGCAUCCUCUGGAGCCUGACCUGUGAUCGUCGCAUCAUAGACCGCCAGUAGACCUGGGAGCAGAUUCACCGCCGCGGCCGU
CUCCUUUAAGACCCCACCGGGUUGGGAGAUGAACUAGAGGAAGUCAUAGUCGAAAAAACGAAACAGACGUUGGCUUCAGUUGCAACAGGUUCAAAAUAUACCCAUUGAUG
CUUUGCAAACUUUAGUAGCCCGGGAGCUUCGCCUGGGUCACAUAGAGCCUUCGCUUAGUAGAUGGAACACUCCGGUGUUUGUAAUUCAGAAGAAAUCAGGGGCUUUUCGC
UUGCUGCAUGACUUGCGCGCUGUCAAUUCUCAACUUAUACCCUUUGGUGUGGUGCAACAGGGAGCACCUGUGUUAUCUGCGGUGCCGGAGGAAUGGGAAGUAACGGCCAU
UGACUUGAAGGAUUGUUUUUUUCUCAAUACCUCUAGCUGAGCAGGAUAGAGAAGCUUUCGCUUUUACAGUAGGACUAGCAAACGGGAGGACUAGCUUACGUGGUGAGAUCC
CUGAGUGGUCUAAGUCCCGAGUACGGGGCAGUCGUCAGUAGUUCAACGCAAUGUGGAGUUGCCUUGAGAUGCUACGUGGACGAGGGCAUGCCCAAGACACACUUUAACCC
UGGCGGGGGUCGCCAGGUGAACUCACCUAAUGGUGUUGGGAUUACAGCCUGAUAGGGUGCUGCAGAGGCCCACGCAUAAGCUAGUAUAAAAAUCUCUGCUGUACAUGGCA
CAUGGAAGU
>0.4-0
AUGUUAUCUUCAGCCGUACAGGGAGCCUUGAACCGAGCCAUUACCGGAGCCCACAAGGUAAUCCGGGUGGUAAUCGCGUGGGUGAUCUCAGCGGUAAUGAAGCUCUAUUA
AAACAGCUCUGGGGUUGUUCCCACUCAAGGGCCCACGUGGCGGCUAGUACUCUGGUAUCUCGGUACCUUUGUACGCCUGUUUUAAUUGCAAAAAUGUGAUCUUGCUUGUA
AAUACAAUUUUGAGAGGUUAAUAAAUUACAAGUAGUGCUAUUUUUGUAUUUAGGUUAGCUAUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGG
AAGCCCUCUCUGCGGUUUUUCAGAUUAGGUAGUCGAAAAACCUAAGAAAUUUACCUGCUACAUUUCAAGGUGCUUGCUAUUUCACCUAAAUCGAAAUAGCGGGUCGUUAG
CUCAGUCGGUAGAGCAGCGGACUUUUAAUCCGUUGGUCGAAGGUUCGAAUCCUUCACGACCCAUGGCGGCGCAUGAGAGAUUAGCCCAAUACCAAAUCAUACUACCCAAA
UAUGGAGAAAGUUCACGUUGACAUCGAGGAGGAUAGCCCCUUCCUCAGAGCUUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACCGAUAACGACC
AUGCUAAUGCUAGAGCCUUUUCGCAUCUAGCUUCUAAGUUGAUCGAAACGGAGGUAGACCCAUCCGAUACGAUCCUAGACAUAGGCAGUGCACCUGCCCGUAGAAUAAUU
AUGGUGGCUCAGAUGGGGCCAACUUCGGUUGGGCCGGUUUCUUCUUGUACCGGUGUUGAGAACCCUGUCUGGGCUACCACCCCUCGGGGCCCAACACCAGGGGAAGCUGU
ACCUUGGUGGUAAGGACUAGAGGUUAGAGGAGACCCCCCGCACAACGGACUCGGCUUGCUAAAGUGCACUUGGCAAGAGGCGAGAGUAGGGAGGUCAGGGCCAGGAGCCC
CCCCCUGAGCCCAGGAUAACUCUUGAAGUUGGGGGGCACUUUCCCGAGGCCACGGCGAGUAGCAUCGAGGGUACAGGAAUGCUGGGAAAAGACUAGGAGAUCUUCUGCUC
UAUUCCAACAUCAGUCACAAGGCACCGAGCGCCGAACACUGUGACUGAUGGGGGAGAAGACCACAGGAUCUCACUGGCAUGCAGCCGUUAUGAUGACGAGUCAGGUCGGG
CUCCGCGGUGGGUUUGGUCGCCCAUGGGACGGAGAUAUGGAAAGGGUCUCGUACGCUGUAAGACUAACUUCGGUUGAGCUUGCGAUAUGGGCCUAUGCCUGGAUAAGUCA
UAAUGUUACUGCCCAGUGGGUGCCUUUGAACCCUUUAUCCCGGGGUUCUUCGGAAGGUGAGACUUGUCCAUGUAGAGUCUCGCCGUGCACGGUAUCACACUGAUGAUACC
UUCAGAGUGCAGGCAUCGCCUACGGUUUUCCGUAGGUUCCCCCUAGGGGCCCCAGUCCCCCAGGUUAUGGGAGAAUAACCCCGACGUGCUGGAAGGGCACAUGGGACGCU
AGUUGUGAACAAGGUGUGAAGAGCCUAUUGAGCUACCUAAGAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACCACGGAGCAAGUGCCUUCAGCCCAGAAGGUAGCUU

GUCGUAACGCGCAAGUCUGUGGCGGAACCGACUACUUUGGGUGUCCCGUGUUUCCCCUUAUUUUCAUUGUGGCUGCUUGCGGGACUAUAAUUGAUUGCCAUCAUAAAGCG
AUUUGGAUUGGCCAUCCGGAGCAUUUUAAGAGGCGUAUAUUAACAAAUAUACUACAGUCGGACGUUUGUGCCAACAUAUAUACUACACUGUACCAAAAAUAUACUACACU
UGCUUCAAAAUAUACUACAACUCCGUUUCAAUAUAACGACACUUUCGUAAAACGGUCUAAAAAUAUGGAAGUAUAUGAGGUAAAUACACGUUUUAAAAACGCUACGAUUU
UUUGUGCCUACUCCUACUUUUUGUAGGGGUAGGCAUUUAGCGAGUAGGACAGACUCUUCAGUCUAGGUUGGAGGAAGCAGGAGAACACUGCUUGAGGGUACAGGUUAGAA
AUUCCUCAGAUUUCUCCCUGUUGACCAGGAGUUAUCGGUGAGUUGCGCUUUGUGCUUUGCGUAAGGGAUUUCUCUCGUACAGGUACGAUCUCGGACCAGCUACGGGUUUG
AAGGGUGAGCCUAUUCUUCGGAAGGGUAAAUUGUAACUUUCACCAAUAAGCAAUGUAGCGCUAACGCAAUUAGUGGAAAAAGAAUUACAGUUAGGACAUAUAGAACCUUC
ACUUAGUUGUUGGAACACACCUGUCUUCGUGAUCCGGAAGGCUUCCGGGUCUUACCGCUUAUUGCAUGAUCUGCGCGCUGUUAAUGCCAAGCUUGUUCCCUUUGGGGCCG
UCCAACAGGGGGCGCCAGUUCUCUCUGCGCUCCCGCAUGGCUGGCCCCUAAUGGUCCUAGACCUUAAGGAUUGCUUCUUCUCUAUUCCUCUUGCGGAACAAGAUCGCGAA
GCUUUUGCAUUCACUAGGCUUGGGCACGGUCCUUCUCGCGGAUAGCGUAGAUGUCUCUAUUCGUUAACAGCUACAUUGUCCGCAGGGAUCUAGCCACAUGGCAAUAGACG
UUCAUCUGCGUGAGCUUCCGAUCUCAGUAUUGUUUGGAAGGAUCAUUUGAUUAACGCGGUUUGAACAGUUUUUUGGAGCUUUUGAUUUCAAAUGUCUAAAAAACCAGGAG
GACCCGGGAAGCCCCGGGUCGUCAAUAUGCUAAAACCAAGGGACAGUGUUGGUGGGACCAACACCUGUCAAUAUAAUUGGGAGAAAUAUUUUAACACAGAUAGGGUGCAC
ACUGGUAUUCCCUAUCAGUUCAAUUGAAACUGUCCAGUGUAAGUGGACUCUUAGUUACUCUGCUCAGCCCACCACUCCACCACACCCAGGCACACACUACACACAGCCAC
CCGUCUCAGCGGGCUAAUACCAGAAGACAAACGUUUACAACUUAAAGACUACUUCAUCAAACAGAUUAGGGCGUACAUGGAUGGUAAAAUACCAGUUGCCACUGUCUCUA
AUGAGUCAAGCGCUUCAAGUAAUACAGUUAAACCAGUUGCGAGUGCAUGGAAACGAAAUAAAUAUGGCACUUACUACAUGGAAGAGAGCGCUAGAUUCACAAACGGCAAU
CAACCAAUCACAGUAAGAAAAGUUGGACCAUUCUUAUCAUGUCCGGCUCGCUCCCUGUAGCCGGAGGGUUAUUUUCCAAGGGUUGAGUCGCAGGACCCCCGGUUCGAGUC
UCGGGCCGGCCGGACUGCGGCGAACGGGGGUUUGCCUCCCCGUCAUGCAAGACCCCGCUUGCAAAUUCCUCCGGAAACAGGGACGAGCCCCUUUUUUUAGGGAAAAUUUG
GCCUUCCAGCAAAGGGAGGCCAGGAAACUUUCCCAGUGCUUUCCAGUUCACUUAAAUAGAACUGGGCCCCCGCAAGGAGGAACGUGUCACUGGCAUGUUCCUGGGAGUUA
ACGGCUCUCCGGCCCCAGCGGUAAGCAAGACGGAUCCUGGGAAACAGGUUUAACGGGCUGGGGGUGGUGCCCCGGCCGUCGCAUCACUUGCUGAUACAAAAGCUACUGUC
CCAGGGGGGGAGGGCCGCCACUUGUGGCAGAUACCUCUUUAACUGGGCAGUAAGGAUGAAUGACCAACGGAAAGAGGCGAAAAACACGCCAUUCAAUAUGAAUUUUAGGA
UACAUACAGUUCAAGAGCAAGCAACGGAUUGAGCCUGUAUGUCUCCUGAUUCAAAGUCCGGCCACAGGUUUUGUGUCACUAAUAUGCAGUGCUGCCUGAGCAAAGACAUG
CCCCAGGUGGACUGGGUAGCCAGGAUCUAAAAAGAAAUAUAGGCUAAAACAUUUAAUAUGGGCAAGCAAAGAGCUGGAAAGAUUCGAAUGUAAUCCUGGUCCCAAUGUCA
GACCACGCCAUGGCGUGCCACUCUGCUGAGAGUGCAGUUGACGACAGUGCCCCAGGAGGACUGGGUUUAGCUCGCCAGUUAGCGAGGUCUGUCCCCACACGACAGAUAAU
CGGGUGCAACUCCCGCCCCUUUUCCGAGGGUCAUCGGAACCAGCCAGACGUGGACCCGGCCUGAUGAGUCCGAAAGGACGAAACAGUAGGUAGUGGAAGAGGUCGGAACU
GCCCACCGCCUUGGUUAUGACGGGUUUCUGUCAUACUAUAGCGGCGCGAAGUUGCGUACUUAUUCGAAUGCC
>0.4-1
AUGUCAUCUUCAGCCAAUUCGACAGCCUGAGAGUCAGCUCUAACCAGAGCCCACAAGGUAAUCUGGGGAGUAAUGAACCCGGUAAUGUCGGUGGUAAUGAUGCUCUGUUA
AAACUGGAUCUGGGUUGUUCCCACCCAGAUCACCUACAUGGUGUUGUACACUAUUAUUACGGUAAUCUUGUACGCCAGUUUUAUACCCAACAAUGUGAUCUUGCUUGCGG
AGGCAAAAUUUGCACAGUAUAAAAUCUGCAAGUAGUGCUAUUGUUGGAAUCACCGUACCUAUUUAGGUUUACGCUCCAAGAUCGGUGGAUAGCAGCCCUAUCAAUAUCUA
GGAGAACUGUGCUAUGUUUAGAAGAUUAGGUAGUCUCUAAACAGAACAAUUUACCUGCUGAACAAAUUGGUGCUUUUAUGUUCACCGAAAUCGAACAUACGCUGAUUUAG
CUCAGUAGGUAGAGCACCUCACUUGUAAUGAGGAUGUCGGCGGUUCGAUUCCGUCAAUCAGCAUUGCGGCGUAGUAUACACUAUUGAAUCAAACAGCCGACCAAUUGCAC
UACCAUCACAAUGGAGAAGCCAGUAGUUAACGUAGACGUAGACCCGCAGAGUCCGUUUGUCGUGCAACUGCAAAAGAGCUUCCCGCAAUUUGAGGUAGUAGCACAGCAGG
UCACUCCAAAUGACCAUGCUAAUGCCAGAGCAUUUUCGCAUCUGGCCAGUAAACUAAUCGAGCUGGAGGUUCCUACCACAGCGACGAUUUUGGACAUAGGCAGCGCACCG
GCUCGUAGAAUAAUUAUGGUGGCUCAGGCAGGGCAGCUUUCGGGCUGGCCGGUAGCUGAUUGGGCCGGUAUUCCUACCCUUGUCUGGGUCACCACCAACCACGGCCCAGC
CUGACUGAAGCUGUAGGUCAGGGGAAGGACUAGAGGUUAGUGGAGACCCCGUGCCACAGGGACUCGGUCUGCUGCAGCGCGCGUAGCAGAAGGCGAGGCUAGGGGAGUCA
GGGUUCGGACCCCCUCCCCGAACCUAGGGUAACACUGACUGUGGAGGGGACAGUGCCGGGGCCACGCGGCGUACGAUCGUGGGUACAGCACUACUGGGACAGACCAGAAA
UCCAACUAUCUACUCAGCAUCAUUCUAGGCACAGAACGCCAGAAAAUGGAAUGGUGCUGUUGAAUCAACAGGUUCUCGUAUUGUUCGGUCUACAUAAUGACGAGUCAGGU
CAGGCUCCGCACUAGGUUCGGUCACCUAGGGGAUGGAGACAUGGAAGGGAUCUCGUGCGCUAUAAGUCUAACGAAAGUUGAGCUUGCGACAUGGGUCUAUGCCUGGAUAA
GUUGUGUAGGUUGCCGCAUACGGGUGCCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACUUCGGUCCGUGUACUUCUAGCACAAGAUGCUAGUUUCAGGGUACGGGC
ACCCCCCCCCUCAUUUAAUGGGGGGCCUCCUCAAGGAGCCCCAGUUCCUAGGUCCAGGAAGGAUCCCGUGGCGCUGAAAGGGCGAUGGGACGCUACAAUACUGACAUGGUG
CGAAGAGUCUAUUGAGCUAGUUGGUAGUCCUCCGGCCCCUGAAUGCGGCUAAUCCUAACUGUGGAGCAGAUACCCACGCACCAGUGGGCAGUCUGUCGUAAUGGGCAACU
CCGCAGCGGAACCGACUACUUUGGGUGUCCGUGUUUCUUUUUAUUCCUACGCUGGCUACUUAUGGUGACAAUUGAGAGACUGUAACCAUAUUGCUAUCGGGUUGGCCACC
UGGUGCACUUUUUGACGCGUAUAUUUACCAAUAUACUACAUUCUGAUGUCUGUAUCUACAUUUUUGGCAAAAUCGACGAGUCAUAUACUACGAUUGCACACAAAUAUACU
ACACUGACCCCUCAAAUAUAUUACGAUCGCGAAAAACGGCCAAAAUAUAUUGAAGUAUAUGGGGCAAAUAUACAAAAAAACAACUCUACGAUUUUCUAGGCCUACUCCUG
CUCAGCGUGGGAGUAGGCCUCUAGCGAGUAAGACAGGCUCUUCAGUCUGAGUUCGCGGAAACGAGUGUGAAUCUGACAACGUAUGCAGGUUAAAAUUCUCCAGGAUUUCU
CAACCUUGGUUGUGCUAUCUGGUGACUUGCGCGUGUUGUCUGCGUAGAGAAUUUCUCUCCUUAACCAAAAGGGGUUUGAAGACUGGGUCUACCGCUUGCGGGAUAAAUUG
UAACUUCCAGUAAACGACGACAUGUUGAUGCUUUGCAAACUUUAGUAGCCCGGGAGCUUCGCCUGGGUCACAUAGAGCCUUCGCUUAGUAGAUGGAACACUCCGGUGUUU
GUAAUUCAGAAGAAAUCAGGGGCUUUUCGCUUGCUGCAUGACUUGCGCGCUGUCAAUUCUCAACUUAUACCCUUUGGUGUGGUGCAACAGGGAGCACCUGUGUUAUCUGC
GGUGCCGGAGGAAUGGGAAGUAACGGCCAUUGACUUGAAGGAUUGUUUUUUCUCAAUACCUCUAGCUGAGCAGGAUAGAGAAGCUUUCGCUUUUACCCAGGCUUGGGCGU
AUUUUCUCCGGUGAGCGUAGCUGCCUGUAGUCGUUAGCAGCUACGUUGUCCGCGGGAUUCUAGCCACCUGGCAAUGAGGAAACAGAUUUCUUUUUUGGAGGAUAACAACU
UAACUUGACUGCGAACAGUUUUUUAGCAGGGAAUUACCCAAUGUCUAAAAAACCAGGAAAACCCGGUAGAAACCGGGUUGUCAAUAUGCUAAAACGUAGGUUCUGUAUUA
GUAGGACCUACCCCUGCCAACAUAAUUGGAAGAAAUGUGAUGACCCAGCUUGGCUUUACUUUAAAUUUUCCCAUUAGUCCUAUUGAAACAGUCCAGUGUAAGUGUCUUUA
AUUUUUAAGCCUCAUGCACAGCACACAAACACCAGGCACACAAACAGCAGCCACACUUCCACACGCGCCCGUCUCCAGAGGUUUAUUACCUAAAGACAAACAGUUACAAU
UAAAAGAUUAUUUCAUUAAACAAAUCAGACAAUAUAUGGAUGGUAAAAUACCUACUGCAACAGUUGUUAAAAGUACAAGUGCUUCUAGUAACACUGUUAAACCAGUAGCA
UCUGGUUGGAAACGUAAUAAUUACGGCACUUAUUAUAAAUCAGAAAAUGCAACUUAUAGCAAUGGUAACACACCAAUUAUCACACGUACAGUAGGACCAUUUAGAAGCUG
UCCGGCUCGCGCCCGUAGUCUGGAGAAGCAUCGCCAGGGUUGAGUCGCGGCAGAACCCGGUUCAAGGACGGCCGCGGCAAGCGGGACUUGGUCACCCCGCCGAUUUAAAG
ACCCACAGCCAGCCGACUUCUCCAGUUACGGGAGCGAGCCCCCUUUUUUAAAGAAGAUCUGGCCUUCCCACAGUGGAAGGCCAAAGAAUUUUUUUAGUGUUUUCGGCUCC
ACUUAAAUCGAAGCCCCCACCCGCAAGGGGGGAGAUUUCCAACUCCCCGGGUGUUGACGACACCCCGGCCCCAUUGACGUACAAGCCGGAUCCUGGGAAACAGGUUUAAC
GGGCUCACUGUGGUGGUGGGCCGUCGAUACACUUGUAUGUGCCCCAAAGCUUAUUGCCCAAGGAGGGAAAGCCGCAGUAUGUGGCAAGUACCUCUUCAAUUGGGCGAUAA
AGAUGAACAACCAACGAAAAAGGACGGGAAAACCGUCUAUCAAUAUGAAUGAGCAACUACAUACAGUUCAAGUCCAAAUGCCGCAUUGAACCUGUCUGUCUUCUGCUCGG
GAGUCAGACCACUUAGUGCCACCAGUAUGAUGAUAAGCUGGUGCUGUCUGUCCUCAUCCCCAGGAGGACUGGGUCGCCAGGGGGAAAGAAAAAAUAUAAAUUAAAGCACA
UAGUCUGGGCAAGCAGGGAGCUAGAACGAUUCGCAGUUAACCCUAGCCCCCGUGUCAGGUCACAUUGUGCCACCUUGCCAGAAGGUGCAACCUGCGAGGCCCCAGGUGGA
CUGGGUUCAGUCGCCAGUGCGAGGCCCAUUCUCUCUCAAUGGGUAUUGGGUGCAACCCCCCCGUCCAUCUCGAACGUCAUCGAGACCAUGUCCGUAGUGGAUGUGUAUCC

ACUCUGAUGAGUCCGAAAGGACGAAACGGAUGAGUUUGUGAGAAGGUUGGGGUCGCCCACCGACUUGGGUAUGAUGGAUUUCUGUCAUACUACAACGGUGCGAAACUCCG
UACUUACACACGGGCU
>0.45-0
AAUUAUGGUGGGCUGGGCAGGGGAGCCGAAAGGCUCGCCGGUUUCCUUGAGCGCCGGUAGUUCCAACCCUGUUCAGUCCGCCACCAGCGGCCAGGGGGAAAGAAAAAAUA
UAGAUUAAAGCUUCUAGUAUGGCCAAGCAGGGGACUAGGCCGAUUCGCAGUUAAUCUUGCCUCAGUUGUCAGACCACACGCAAGUGUGCUACUCUGCGAAGAGUACUGUC
UGCGUUAGAGCCCCAGGAGGACUGGGAUGGUGCCUACUCCUACUUUCUGUAGGGGUAGGCAUUUCCCCAGUCCCCCAGGUUAUGGGAGAAUAACCCCGACGUGCUGGAAG
GGCAGUAAGUGGACUUUCAUUUAUUCCGCUCAGCCCACCACUCCACCACACCCAGGCACACACUACACACAGCCACCCAUCUCAGAGAAGUUCAUCUGUGUGAACUUAUU
CCAAACAGCUUUUGGAGUAGUGCGUGUGAACGUAAACACAGUUUGAACGUUUUUUGGAUAGAGACAACUAUGUCUAACAAAAAACCAGGAAGACCCGGCUCAGGCCGGGU
UGUCAAUAUGCUAAAGCAGUGGCUAUCCCUCCACUUAAAUCGAAGGGUGUUGGUUUUUCUCCAAGGAUUUACUGACAACCGAAGCCCUGACUGUUACCGCAGUUGGGGCU
UCAACUCUCCGUGUAGCUCAGUUUGGUAGAGCGCCUGAUUUGGGAUCAGGAGGUCCAAGGUUCAAAUCCUUGUAUGGAGAACGCUGUCUGUACUUGUAUCAGUACACUGA
CGAGUCCCUAAAGGACGAAACAGCGCAGCUAGUAGGACAGACUCUUCAGUCUAGGUUGGUGGAAGCAGGAGAACACUGCUUGAGGGUACAGGUUUGGAAGCCGCGGCCAC
GCCGAGUAGGCACGAGGGUACAGCUUCCGACUCCUCUCCGUAGUCUGGAGGUUAGGUCGCAAGGGUGCGGCGGCGGGGAACCCUGGUUCGAAACCAGCUGGAUCCGCCGU
CCCGAUGCGCUUGGCCCCGCAUCCACGACGGCCCCGGGCGUCGAGACCCAGCCGCGAUACGCACACCCCAAAUACGGAGGGGAGUCUUUUAGCGCUAACGCAAUUAGUGG
AAAAAGAAUUACAGUUAGGACAUAUAGAACCUUCACUUAGUUGUUGGAACACACCUGUCUUUGUGAUCCGGAAGGCUUCCGGGUCUUAUCGCUUAUUGCAUGACUUGCGC
GCUGUUAACGCCCAGCUUGUUCCUUUUGGGGGCCGCCAACAAGGGGCGCCAAUUCUCUCCGCGUUCCGGCGUGGCUGCCCCCUGAUGUUCUUAGACCUCAAGGGUUGCUU
CUUUUUUAUUCCUCUUGCGGAACAAGAUCGCGAAGCUUUUGCAUUUACAUGAAUAACCAACGGAAAAAGGCGAAAAACACGCCUUUCAAUAUGCACUUUCAGCGACGUAU
AUUGACCAAUAUACUACAAUCGGACAUUUGCACCGAUCUAUACACUACGAUUUACAAAAAAUAUACUACACUAGCACGCAAAUAUACUACAACUCCGUUUCAAUAUAACU
ACACUUUCGUAAAACAACCCCAAAAUAUGGUAGUAUAUGCUGAAAAUACACAAUUAAAAAACGCUACGAUUUCUGGGCGGUAAGCAAGAGGGAUCCUGGGAAACAGGCUU
CGACGGGUUGGGGGUGGUGCCCCGGCCGACGCAUCACUUGCUGAUACAACCAGCGGACGGACUAGUACUUGUACUACAGAAUUAACUGCCGUGUGCCGCCCGCUAAACUA
GCCCCAAUCAUCGAAAAUGGAGAAACCGACAGUGCACGUUGACGUAGACCCCCAAAGUCCGUUUGUGCUACAACUGCAGAAGAGUUUCCCACAAUUCGAGAUUGUGGCUC
AGCAGGUCACUCCGAAUGACCAUGCUAAUGCCAGAGCUUUUUCGCAUCUGGCUAGUAAACUGAUCGAACAUGAGAUCCCCACCUCAGUUACGAUCUUGGACAUAGGAAGC
GCACCAGCUCGUAGAAUAAUGAAGGCAGUCCUUCUCCCUUUAGCACACAAAGGUCAAUGGGUGCGACUCCCCCCCUUUUCCGAGGGUCAUCGGAAACCCGACCUACUGCG
GCAGCACACACAGUGACGGGAAGUUGGUCGCUCCCGACGCAGUUAGGUCAGAGAAAAUUUUGUGAGACCAAGCAAAAAUGUGAUCUUGCUUGUAAAUACAAUUUUGAGAG
GUUAAUAAAUUACAAGUAGUGCUAUUUUUGUAUUUAGGUUAGCUAUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGGAAGCCCUCUCUGCGGU
UUUUCAGAUUAGGUAGUCGAAAAACCUAAGAAAUUUACCUGCUACAUUUCAAGCGCCACUAGUGCAUCGAAUCCUGGGAAACAGGCACGGAGAACAAAAUCCGGUAAGCU
CGGGAAGCCGUCAACUUCCGCCGUAUUGAUGAAAACAACUGGUAGUGGAAGAGGUCGGAACUGCCCACCGCCUUGGUUAUGACGGGUUUCUGUCAUACUAUAGCGGCGCG
AAGUUGCGUACUUAUUCGAAUGCCCUGGGAGAGACCAGAGAUCCUGCUGUCUCUACAGCAUCAUUCCAGGCACAGAGCGCCAGAAAAUGGAAUGGUGCUGUUGAAUCAAC
AGGUUCUCACAGUCAGGCCGAAAGCCACCACAAGCGGUACAGUGGUGCUGCCUGCAUUAACCGCCCCAGGUGGACUGGGACCACCCGCAAGGGGAGGGUUUUCUAACUCU
CCGGGUGUUGACGACACCCCGGCCCCAUUGGGGCCCAAGGUGUGUUGAAGCUGUAGCCACACUGGAAGGACCAGAGGUUAGAGGAGACCCCCCCAGACAAGGACUCGGCU
UGCUGAAGUGCACUCGGCAAGAGGCGAAAGUUUUUUAGGGAAAGUUUGGCCUUGGAGCAAGGGAAGGCCAGGGCAUUUCCUUCAUAAAUAAUUACGUACAGUUCAAGAGC
AAGCACCGUAUUGAGCCAGUAUGUUUGUUAGUGGGUACCCUUGAAAUCACCUCCUAGAUUUCUUCGGAAGGGCUUCGUGAGAAGCUCGUGCACGGUAAUACACUGAUAUU
ACCAAGAGUGCGGGUAUCGCCUGUGGUUUUCCACAGGUUCUCCAUAAGGAGGAGUGUUCCCACCCAACAGGCCCACUGGGUGUUGUACUCUGGUAUUACGGUACCUUUGU
ACGCCUAUUUUAUCUCUCAUCGUCCGGGGGGGGAGGGCGGCCACCUGUGGCCGCUACCUCUUUAACUGGGCGGUGAAGCACUGGCAUGCAGCCGUUAUGAUGACGAGUCA
GGUCGGGCUCCGCGGUGGGUUUGGUCGCCCAUGGGACGGAGAUAUGGAAAGGGUCUCGUACGCUGUAAGACUAACUUCGGUUGAGCUUGCGAUAUGGGCCUAUGCCUGGA
UAAGUCAUAAUGUUACUGCCCAGUG
>0.45-1
GUCAAUGAUGAACUGGUUGAGGGAUCGAAAGAUCGCCGUUUACACCGUGUUACGGUACGCCAACCUUGAUCAGUUCAUCACCAGUGACCUGGGGGAAAGAAAAAGUAUAU
GAUGAAACAUUUAGUUUGGGCAAGCAGGGAGCUGGAAAGAUUUGCUUGUAACCCAGGUCCCCGUGUCAGGUCACAUUGUGCCACCUUGCCAGAAGGUGCAACCUGCGAGG
CCCCAGGUGGACUGGGUCUAGGCCUACUCCUGCUCAGCGUGGGAGUAGGCCUCUCCCCAGUUCCUAGGUCCAGGGAGGGCCCCGUGGCGCUGAAAGGGCGGUAAGUGUCU
UUAAUUUUUAAGCCUCAUGCACAGCACACAAACACCAGGCACACAAACAGCAGCCACACUUCCACACGCGCCCGUCUCCAGGAGGAAACAGAUUUCUUUUUUGGAGGAUA
ACAACUUAACUUGACUGCGAACAGUUUUUUAGCAGGGAAUUACCCAAUGUCUAAAAAACCAGGAAAACCCGGUAGAAACCGGGUUGUCAAUAUGCUAAAACAGUGCUUUC
CAGUUCACUUAAAUAGAACUGGUCGUUAAUUCCUAUGCGUGGUUUUACGAUACCGAAGCCCUGACUGUUCCCGCAGUUGGGGCUUCAACUGCUGAUUUAGCUCAGUAGGU
AGAGCACCUCACUUGUAAUGAGGAUGUCGGCGGUUCGAUUCCGUCAAUCAGCAGAAGAGUCGUGCUUAGCACACUGAUGAGUCUCUGAAAUGAGACGAAACUCUUUGGUG
AGUAAGACAGACUCUUCAGUCUGAGUUUGUGAAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUACAGUGCCGAGGCCACUCGGAGUACGAUCGAGGGUACAGCACUAG
GCACUCUUCCGUGGUCUGGUGGAUAAAUUCGCAAGGGUAUCAUGGCGGACGACCGGGGUUCGAGCCCCGUAUCCGGCCGUCCGCCGUGAUCCAUGCGGUUACCGCCCGCG
UGUCGAACCCAGGUGUGCGACGUCAGACAACGGGGGAGUGCUCCUUUUGAUGCUUUGCAAACUUUAGUAGCCCGGGAGCUUCGCCUGGGUCACAUAGAGCCUUCGCUUAG
UAGAUGGAACACUCCGGUGUUUGUAAUUCAGAAGAAAUCAGGGGCUUUUCGCUUGCUGCAUGACUUGCGCGCUGUCAAUUCUCAACUUAUACCCUUUGGUGUGGUGCAAC
AGGGAGCACCUGUGUUAUCUGCGGUGCCGGAGGAAUGGGAAGUAACGGCCAUUGACUUGAAGGAUUGUUUUUUCUCAAUACCUCUAGCUGAGCAGGAUAGAGAAGCUUUC
GCUUUUACAUGAUCAACCAACGGAAAAAGACGGGUCGACCGUCUUUCAAUAUGCACUUUUUGACGCGUAUAUUUACCAAUAUACUACAUUCUGAUGUCUGUAUCUACAUU
UUUGGCAAAAUCGACGAGUCAUAUACUACGAUUGCACACAAAUAUACUACACUGACCCCUCAAAUAUAUUACGAUCGCGAAAAACGGCCAAAAUAUAUUGAAGUAUAUGG
GGCAAAUAUACAAAAAAACAACUCUACGAUUUUUUGACGUACAAGCCGGAUCCUGGGAAACAGGUUUAACGGGCUCACUGUGGUGGUGGGCCGUCGAUACACUUGUAUGU
GCCCCAAAUGGGCGGCGCAUGAGAGAAGCCCAAAUCAAUAACUACCCAAAAUGGAGAAAGUUCACGUUGACAUCGAGGAAGAUAGUCCCUUCCUCAGAGCAUUACAACGG
AGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACAGAUAAUGACCAUGCUAACGCCAGAGCGUUUUCGCAUUUGGCAUCGAAAUUGAUCGAGACGGAGGUGGAACC
AUCCGAUACGAUCCUAGACAUUGGAAGUGCGCCUGCCCGCAGAAUUCAGGUCGCCAGUGCGACACCCGUUCCCACACAACGGGCAUUGGGGUGCAACUCCCCCGUCUAUC
CUGAACGUCAUCAGGACCAAGGGGGUGAUGUGGCAGCGCACCACGACAUCGUGACGGGAAGAGGUCGUCCCCCACGCAUCAUCUCUCUAGGGCAUUUUCGUGAGACCCCA
ACAAUGUGAUCUUGCUUGCGGAGGCAAAAUUUGCACAGUAUAAAAUCUGCAAGUAGUGCUAUUGUUGGAAUCACCGUACCUAUUUAGGUUUACGCUCCAAGAUCGGUGGA
UAGCAGCCCUAUCAAUAUCUAGGAGAACUGUGCUAUGUUUAGAAGAUUAGGUAGUCUCUAAACAGAACAAUUUACCUGCUGAACAAAUUACACCACUCGCAAAUACGGAU
CCUGGGAAACAGGCAGAACUUCGGUUCGUAAGCUCGGGUAAACUAGUCCUUUACCGCCGUAUCGUACUUUGUGUGGCCAGUUUGUGAGAAGGUUGGGGUGGCCCACCGAU
AUGGGUAUGAUGGUUUCCUGUCAUACUACAGUGGUGCGAAACUCCGUACUUACACACGAGCUCUGGGAUAGACUAGGGGAUCUUCUGCUCUGCACAACCAGCCACACGGC
ACAGUGCGCCGACAUAGGUGGCUGGUGGUGCUAGAACACAGGAUCUGAUAGUCAGGCCAGGCAAUGCCUGCCACCGGAAGUUGGAUGACGGUGCUGUCUGCGCUCCAACC
CCAGGCGGACUGGGUGCUCCCGUAAGGAAAGCGCAAGCUUUGAGCAUUGACAACGCUCCGGCCCCACACGGCCCAGCCUGGCUGAAGCUGUAGGUCAGGGGAAGGACUAG
AGGUUAGUGGAGACCCCGUGCCGCAAGGACUCGGUCUGCUGCAGCGCGCGUAGCAGAAGGCGAGGCUUUUUUAGGGAAGAGCUGGUCUCCCUUCAAAGGGAGACCAGGAA

ACUUCCCCAAUGAGCAACUACAUACAGUUCAAGUCCAAAUGCCGCAUUGAACCUGUCUGUCUUCUGCUCGGCACCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACU
UCGGUCCGUGUACUUCUAGCACAACGUGCUAGUUUCAGGGUACGGGUGCCCCCCCCAUUUUCGUGGGGGCCUCCAAAAGGAGUUAAAACAGCUCUGGGGUUGUUCCCACCC
CAGAGGCCCACGUGGCGGCUAGUACUCCGGUACCCCGGUACCCUUGUACGCCUGUUUUAUACAGCUUAUUGCCCAAGGAGGGAAAGCCGCAGUAUGUGGCAAGUACCUCU
UCAAUUGGGCGAUAAAGUCAUGGGUUCCUUCCCAUACGAUGACGAGUCAGGUCGGGCCCUAUCUUAGGUUUGGUCACCUAGGGGACGGGGAUAUGGAAAUCACUUUCGCU
UGCUGUCAGUCUAGUGGAAACACUUAGCUUGCAAUGUGGGUGUAUGCCUGGAUAAGUCGUAUGGAUGCUGGCCAUGA
>0.501-0
GCCAGACGUGGACCCGGCCUGAUGAGUCCGAAAGGACGAAACAGUAGGUAGUGGAAGAGGUCGGAACUGCCCACCGCCUUGGUUAUGACGGGUUUCUGUCAUACUAUAGC
GGCGCGAAGUUGCGUACUUAUUCGAAUGCCGGAGACCGCGGCCACGCCGAGUAGGAACGAGGGUACAGUCUCUGGGAUCGGCUUGCUGAAAGUGCACUCGGCAAGAGGCG
AGAGUGGAGAUGGAUGCUGGAUCCUGGAAAACAGGUUUAACGGGUAGGGGGUGGUGCCCCUACCGACGCAUCACUCCAGAUACCAAGCGAGUAAGACAGACUCUUCAGUC
UGAUGUUACGGAGAUGAGUAGAACAAAGUGAGGGACUAUGGGUUAAAGCCUAAUGGCCAUAGUAGGAGGGGUAAUCUAUCUGGGGAAGUUGGUGGUUCUUCUUCCUCCUU
GCCCUCAGGUGCACAGACUGGGGAGUGGAUAGAUAAUGACUAUGGAGACAGCUUAUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACAUUCCGAAGG
GGACCGUCCCUCGGUAAUGGCGAAUGGGACCCAGGUGCCUUUGAACCCUUUAUCCCGGGGUUCUUCGGAAGGUGAGACUUGUCCAUGUAGAGUCUCGCCGUGCACGGUAU
CACACUGAUGAUACCUUCAGAGUGCAGGCAUCGCCUACGGUUUUCCGUAGGUUCCCCCUAGGGGAACCCCGUAAGGAAGGACAAGGCUGUCCUUGAGUACUAACGACACU
CCGGCCCCAUAAUUGAGGACAGUUCCUCUCCCUCUAGCACACAGAGGUCAAACUGGGUGCAACUCCCCCCCCUUCCGUGGGUAACGGAAACCAGACGUUCAUCUGCGUGA
GCUUCCGAUCUCAGUAUUGUUUGGAAGGAUCAUUUGAUUAACGCGGUUUGAACAGUUUUUUGGAGCUUUUGAUUUCAAAUGUCUAAAAAACCAGGAGGACCCGGGAAGCC
CCGGGUCGUCAAUAUGCUAAAACUUUUUUAGGGAAGAGCUGGUCUCCCUUCAAAGGGAGACCAGGAAACUUCCCCCCCCAGUCCCCCAGGUUAUGGAGGAAUAACCCCGA
CGUGCUGAAAGGGCAUGUUGGGUUCUAGAUGUUAUGAUGACGAGUCGGUCGGCUCCGCGCUAGGUUUGGUCGCCUAGGGGAUGGAGAUAUGGAAAGGGUCUCGUGUGCUG
UCAGUCGGUCGAAAGACGCGCUUGCAACAUGGGCCUAUAACCGGAUAAGUCAUAGCAAUACUAGCCAACACAACGUCAGGCCACAAUUCUGUGCCACUCCGCUGGGGAGU
GCGGCCUGCGUAGCCCCAGGAGGACUGGGUGCGGACGGACUAGUACUUGUACUACAGAAUUAACUGCCGUGUGCCGCCCGCUAAACUAGCCCCAAUCAUCGAAAAUGGAG
AAACCGACAGUGCACGUUGACGUAGACCCCCAAAGUCCGUUUGUGCUACAACUGCAGAAGAGUUUCCCACAAUUCGAGAUUGUGGCUCAGCAGGUCACUCCGAAUGACCA
UGCUAAUGCCAGAGCUUUUUCGCAUCUGGCUAGUAAACUGAUCGAACAUGAGAUCCCCACCUCAGUUACGAUCUUGGACAUAGGAAGCGCACCAGCUCGUAGAAUAGUGG
CUAUCCCUCCACUUAAAUCGAAGGGUCUGGGAAAGACCAGAGAUCCUGCUGUCUCCCCAGCAUCAUUCCAGGCACAGAACGCCAGAAAAUGGAAUGGUGCUGUUGAAUCA
ACAGGUUCUAUGUCUAAGAAACCAGGAGGGCCCGGUAAAAACCGGGCUGUCAAUAUGAAUUAUGGUGGGCUGGGCAGGGGAGCCGAAAGGCUCGCCGGUUUCCUUGAGCG
CCGGUAGUUCCAACCCUGUUCAGUCCGCCACCAGCGGCACUCUUCCGUGGUCUGGUGGAUAAAUUCGCAAGGGUAUCAUGGCGGACGACCGGGGUUCGAGCCCCGUAUCC
GGCCGUCCGCCGUGAUCCAUGCGGUUACCGCCCGCGUGUCGAACCCAGGUGUGCGACGUCAGACAACGGGGGAGUGCUCCUUUUAAAACUGGAUCUGGGUUGUUCCCACC
CAGAUCACCUACAUGGUGUUGUACACUAUUAUUACGGUAAUCUUGUACGCCAGUUUUAUACGCGAGAAUGGUCAAAUUGGUAAAGGCACAGCACUUAAAAUGCUGCGGAA
UGAUUUCCUUGUGGGUUCGAGUCCCACUUCUCGCAACACUACAUAUUUAUUUAUUUAUUUCUUAGUAAUGUUUACUGGAACUUAAAUCUGUGAUAACCUAAACUAAAAGU
CUCAAACAACCCGUUACUUGCUGACCAAUUUUUGUAGGUACUGGGUGUAAAUAUGAUGACCGGUACCAAGUUGUUAGUCUGUGUGGACCGACAAGGACAGUUCCAAAUCG
GAAGCUUGCUUAACACAGUUCUAACAGACACCACUCGCAAAUUCGGAUCCUGGGAAACAGGCAGAACUUCGGUUCGUAAGCUCGGGUAAACUAGUCCUUUACCGUCGUAU
CGUACUUUGUGUGGCCUGGGGGCCCAAGGUGUGUUGAAGCUGUAGCCACACUGGAAGGACCAGAGGUUAGAGGAGACCCCCCCAGACAAAGGCUUCUGGCCAGAGGAGGC
AAGGCUGCCAUAUGUGGCAAGUACCUCUUCAACUGGGCAGUAAGAAAUUUUAGGAUACAUACAGUUCAAGAGCAAGCAACGGAUUGAGCCUGUAUGUCUCCUGAUU
>0.501-1
CCGCUAUAUGGGGAUGUGUGUCCCUACUGACGAGUUCAAAAGAACGAAAUAGUUAGGUUUGUGAGAAGGUUGGGGUAGCCCACCGACUGUCAUACAGCGGUGCGAAACUC
CGUACUUACACACGAGCCCAGUGCCGAGGCCACGCGGAGAACGAUCGAGGGUACAGCACUAGGACUCGGUCUGCUGCAGCGCGCGUAGCAGAAGGCGAGGCUUGACGUAC
AAGCCGGAUCCUGGGAAACAGGUUUAACGGGCUCACUGUGGUGGUGGGCCGUCGAUACACUUGUAUGUGCCCCAAAGCGAGUAGGACAGACUCUUCAGUCUAGGUUGGAG
GAAGCAGGAGAACACUGCUUGAGGGUACAGGUUAAAGCCUAAUGGCCUAGAUCGGAGGGGUCGUGGUAAUAGCACCGUUGGUGGUUCUUCGGAUGCUUCUCAGUCCCAAG
GCGCACAAGCAUGGGGGGGUUACCAUAACGAUCUAGGAGACAGCUUUCGGGUCGGCAUGGCAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUCCGAAGGAGGACGCACG
UCCACUCGGAUGGCUAAGGGAGAGCCACUGGCACCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACUUCGGUCCGUGUACUUCUAGCACAACGUGCUAGUUUCAGGG
UACGGGUGCCCCCCCCAUUUUCGUGGGGGCCUCCAAAAGGAGUUCAGCGCAAGGAAGGAAAGCUGGACGCUACCUUAGGUGGUGACGACACCUCGCCCCCAUCAGCUCGCC
AGUUAGCGAGGCCUGUUCCCACACAACAGGUAAUCGGGUGCAACUCCCGCCCCUACUCCGAGGGUCAUCGGAACCAAGAAGUUCAUCUGUGUGAACUUUGUGAUUGACAG
CUCAACACAAGUGCGGACUACCGUAAACACAGUUUGAACGUUUUUUGGAGAGAGACUACUAUGUCUAACAAAAAACCAGGAAAACCCGGGGCAACCCGGGUUGUCAAUAU
GCUAAAGCUUUUUUAAAGAAGAUCUGGCCUUCCCACAGUGGAAGGCCAAAGAAUUUUUUUCCCCAGUUCCUAGGUCCAGGUAGGAUCCCGUGGCGCUGUGAGGGCGUGUC
AGGUUGGAACCCACACUAAUUGAGCUAGGCCGGGCCCCGUCGUAGGUUUGGUCACCUACGGAAUGGGGAUAUGGAAAGGAUCUCGUACCCUGUGAGUCAAACGAAAGUUG
UGCUUGGGAUAUGGGUCUAUGCCUAGUAGGUUAUGUGGUGUGUCUCUGACUACAAGUCAGACCAGAAAUGCCACCUGAAAGCAUGCUAAAGGUGCUGUCUGUACAUGCCC
CAGGAGGACUGGGUAUGGGCGGCGCAUGAGAGAUUAGCCCAAUACCAAAUCAUACUACCCAAAUAUGGAGAAAGUUCACGUUGACAUCGAGGAGGAUAGCCCCUUCCUCA
GAGCUUUACAACGGAGCUUCCCGCAGUUUGAGGUAGAAGCCAAGCAGGUCACCGAUAACGACCAUGCUAAUGCUAGAGCCUUUUCGCAUCUAGCUUCUAAGUUGAUCGAA
ACGGAGGUAGACCCAUCCGAUACGAUCCUAGACAUAGGCAGUGCACCUGCCCGUAGAAUAGUGUUUUGAAGUCCACUUAAAUAGAACUUCCUGGGAUAGACUAGGAGAUC
UUCUGCUCUGCACAACCAGCCACACGGCACAGUGCGCCGACAAUGGUGGCUGGUGGUGCGAGAACAAUGAACAACCAACGAAAAAAGACGGGAAAACCGUCUAUCAAUAU
GAAUUAUGGUGGGGCGUACAGGGGCAUCGCAAGAUGCGCCGGUAAGGGUAGCCGCCGGUAACGCCAACUCUGUACGUCUCACCACCUCUGGCUCGACUCCGUGGCCUGGA
GGCUAAGCGAACGGGUUGGGCUGCGCGUGUACCCCGGUUCGAGUCCCUGCUCGAAUCAGGCUGGAGCCGCAGCUAACGUGGUACUGGCACUCCCGUCUCGACCCAAGCCU
GCUAACGAAACCUCCAGGAUACGGAGGCGGGUCGUUUUAAAACAGCCUGGGGGUUGUACCCACCCCUGGGGCCCACGCGGCGCCAGUACUCUGGUACGCUAGUACCUUUG
UACGCCUGUUUUCCCCUGGGAAUUAGCCAAGUUGGUAAGGCAUAGCACUUUGACUGCUAGAUGCAAAGGUUCGAGUCCUUUAUUCCCAGACACUACAUAUUUAUUGUUUA
UUUAUACCUGAUAAUGCUGCUUUAAUCACACACGAGGUAUCGCUUCUACUAAUUUUUGAAAGCGCUGGGUGUACAUAUUUAAAAACCAGCGCUGAGUUGUUGAUCUGUGU
GAAUCAGACUGCGACAGUUCGAGUUUGAAGCGAAAGCUAGCAACAGUAUCAACAGGCGCCACUAGUGCAUCGAAUCCUGGGAAACAGGCACGGAGAACAAAAUCCGGUAA
GCUCGGGAAGCCGUCAACUUCCGCCGUAUUGAUGAAAACAACUCCACGGCCCAAGCCUCGUCUAGGAUGCAAUAGACGAGGUGUAAGGACUAGAGGUUAGAGGAGACCCC
GUGGAAAUACCUCAUCAGCCGGGGGGGGAGGGCGGCCACGUGCGGUCGCUACCUCUUCAACUGGGCUGUGAAGGAUGAGCAACUACAUACAGUUCAAGUCCAAAUGCCGU
AUUGAACCUGUCUGUUUAUUGCUG
>0.55-0
AGUGUUUCGGCUUCCACUUAAAUCGAAAGCCAUUUAUGGCGGGCUGUGUGGGGGAGCCGAAAGGCUCGCCGGAUGUUUCGACCGGUAGUGCUAACCCCGCACAGUUCGCC
ACCACGAUGGGCGGCGCAUGAGAGAAGCCCAAGAAUAAAACUACCCAUCAUGGAGAAGGUUCACGUUGAUAUCGAGGAAGACAGCCCCUUCCUCAGAGCUUUACAACGGA
GCUUCCCCCAGUUUGAGGUAGAAGCCAAGCAGGUUACAGACAACGACCAUGCUAACGCUAGAGCGUUUUCGCAUCUAGCAUCUAAACUGAUCGAGACGGAGGUGGAACCA
UCCGAUACGAUCUUAGACAUUGGAAGUGCGCCCGCUCGCAGAAUUGUCCGUAGUGGAUGUGUAUCCACUCUGAUGAGUCCAAAAGGACGAAACGGAUGUUAAAACUGGAU

CUGGGUUGUUCCCACCCAGAUCACCUACAUGGUGUUGUACACUAUUAUUACGGUAAUCUUGUACGCCAGUUUUAUACCCGUGUCAGAUCGCGAAAGCGCCACUUCGCCGA
GGAGUGCAAUCUGUGAGGCCCCAGGAGGACUGGGUUGGGAAUUAGCCAAGUUGGUAAGGCAUAGCACUUUGACUGCUAGAUGCAAAGGUUCGAGUCCUUUAUUCCCAGAC
GACGAACUGUGGCAGCACACCAUUUGGUGACGGGAAGCUGGUCGCUCCCGACGCAGGUCGUCGUAAGAACUUUGUGAGACCAAAAGGUACCCUUGAAAUCACCUCCUAGA
UUUCUUCGGAAGGGCUUCGUGAGAAGCUCGUGCACGGUAAUACACUGAUAUUACCAAGAGUGCGGGUAUCGCCUGUGGUUUUCCACAGGUUCUCCAUAAGGAGCCCCAGU
CCCCCAGGUUAUGGAGGAAUAACCCCGACGUGUUGGAAGGACACUGGGAAUAGACUGGGAGAUCUUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGU
AUGUAGCUGGUGGUGAGUAAGAACACAGGAUCUAGAAGUUCAUCUGUGUGAACUUUGUGAUUGACAGCUCAACACAAGUGCGGACUACCGUAAACACAGUUUGAACGUUU
UUUGGAGAGAGACUACUAUGUCUAACAAAAAACCAGGAAAACCCGGGGCAACCCGGGUUGUCAAUAUGCUAAAGCGGCUCGCGCCCGUAGUCUGGAGAAGCAUCGCCAGG
GUUGAGUCGCGGCAGAACCCGGUUCAAGGACGGCCGCGGCGAGCGGGACUUGGUCACCCCGCCGAUUUAAAGACCCACAGCCAGCCGACUUCUCCAGUUACGGGAGCGAG
CCCCCGACAGUUUCUUACUUUAGAAAAAGUAAGUGGUGCGGAUGGGGUUACCCCGCCUGGUUUCUACAUUCCAGUAAAGAUGUAGUGGUGGAGUCAACGAUGGGGGCCCA
AGGCGAGAUGAAGCUGUAGUCUCGCUGGAAGGAAUAGAGGUUAGAGGAGACCCCCCCGAAACACCCUCAUCUCCCAUGGAGGGAGGGCGGCCGUUUGCGGUCGGUAUCUC
UUCAACUGGGCGGUGAAGGCCCGCGCAAGGAAGGAUGCGAUGAAAACGCUUGUCCUUGGGAGUUGACGACUCUCCGCCCCCAGUGGCCGGCAUGGUCCCAGCCUCCCCGC
UGGCGCCGGCUGGGCAACAUUCCGAGGGGACCGUCCCUCGGUAAUGGCGAAUGGGACCCAAGACUCGGCUUGCUGAGGUGCACACGGCAAGAGGCGAGAAGUUUUUAGGG
AAGAGCUGGUCUCCCUCAAAGGGAGACCAGAAACUUCCCCACACCUGACUCCAGACUCUUCAGUCUGAGUUUGUGGAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUUA
GAAAAUGUGUGAUCUGAUUAGAAGUAAGAAAAUUCCUAGUUAUAAUAUUUUUAAUACUGCUACAUUUUUAAGACCCUUAGUUAUUUAGCUUUACCGCCCAGGAUGGGGUG
CAGCGUUCCUGCAAUAUCCAGGGCACCUAGGUGCAGCCUUGUAGUUUUAGUGGACUUUAGGCUAAAGAAUUUCACUAGCAAAUAAUAAUUGGUGAGUUGCUGUUGUGGGA
UCCUGGGAAACAGGUUCGGUGAACAAAAGCCGGUAAUCUCGGGCGUUUGUCAGACGCCGCCGAAUCAACACAGUAUGCAUAUGUCUAAAAAACCAGGAAAACCCGGUAGA
AACCGGGUUGUCAAUAUGUAGGCUUGGGCACGGUCCUUCUCGCGGAUAGCGUAGAUGUCUCUAUUCGUUAACAGCUACAUUGUCCGCAGGGAUCUAGCCACAUGGCAAUU
UAGCUCGCCAGUUAGCGAGGUCUGUCCCCACACGACAGAUAAUCGGGUGCAACUCCCGCCCCUUUUCCGAGGGUCAUCGGAACCA
>0.55-1
AGUGCUUUCCAGUUCACUUAAAUAGAACUGGGUCAAUGAUGAACUGGUUGAGGGAUCGAAAGAUCGCCGUUUACACCGUGUUACGGUACGCCAACCUUGAUCAGUUCAUC
ACCAGUUUCGCCUUUCUUGGAUCCCUUGUUACUCCACCGCAGUGAGAGAUAAACAACCCAGAAUGAAGGUCACUGUAGAUGUUGAGGCUGACAGCCCAUUUUUGAAGGCC
CUUCAGAAAGCAUUCCCGGCUUUUGAGGUUGAAUCACAGCAGGUCACACCUAAUGACCAUGCUAAUGCCAGAGCUUUUUCGCAUCUGGCCACAAAGUUAAUUGAACAAGA
GGUUCCAGCCAACAUCACCAUCCUGGACGUGGGCAGUGCGCCCGCAAGGAGGUUGAUGAGUCUGUGCUAAGCACACUGAUGAGUCUCUGAAAUGAGACGAAACUCAUGUC
CCACCAACAGGGCCCACGGGGCGCCAGCACUCUGGUAUCACGGUACCCUUGUGCGCCUGUUUUAUAACACAGUCAGGCCGAAAGCCACCACAAGCGGUACAGUGGUGCUG
CCUGCAUUAACCGCCCCAGGUGGACUGGGAGGGUCGUUAGCUCAGUCGGUAGAGCAGCGGACUUUUAAUCCGUUGGUCGAAGGUUCGAAUCCUUCACGACCCAAGGGGGU
GAUGUGGCAGCGCACCACGACAUCGUGACGGGAAGAGGUCGUCCCCCACGCAUCAUCUCUCUAGGGCAUUUUCGUGAGACCGGUGCCUUUGAGAGUUACUCUUUGCUCUC
UUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAAGUCUUAGAAUGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGCCCCAGUUCC
UAGAUCUAGGAAAGAUCCCGUGGCGUUGUGAGGACGCCCUCAAUAGAGGGGGGCGGUUCUUGUUCUCCCUGAGCCACCAUCACCCAGACACAGACAGUCUGACAAGGAA
GUGAUGUGUGACUCGGAAAAACACCCGCUGAGGAAACAGAUUUCUUUUUUGGAGGAUAACAACUUAACUUGACUGCGAACAGUUUUUUAGCAGGGAAUUACCCAAUGUCU
AAAAAACCAGGAAAACCCGGUAGAAACCGGGUUGUCAAUAUGCUAAAACGGCACUCUUCCGUGGUCUGGUGGAUAAAUUCGCAAGGGUAUCAUGGCGGACGACCGGGGUU
CGAGCCCCGUAUCCGGCCGUCCGCCGUGAUCCAUGCGGUUACCGCCCGCGUGUCGAACCCAGGUGUGCGACGUCAGACAACGGGGGAGUGCUCCUUCCGAGUCUUGGAAU
GACUUAAAUCUUACCCUGGUGCGGAUGGGAUCUCUCUCCCGCCUGGUUUCUUGCUUCCAGUCAAAAAGCAAGUGGUGGUGCCAAAUGGAGGCCAUGCGCCACGGAAGCUG
UGCGCGUGGCAUAUUGGACUAGCGGUUAGAGGAGACCCCUCCCAUCACAAGCUACUGUCCCAAGGGGGGAGGGCUGCCACCUGUGGCAAAUACCUCUUCAAUUGGGCUGU
AAGGGCCCACGCUAGGGGGUAGAGUAGACUCAUCUCCGGGUGUUGACGACACCCCGACCCCAUCGGGUCGGCAUGGCAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUC
CGAAGGAGGACGCACGUCCACUCGGAUGGCUAAGGGAGAGCCACUGGACUCGGUCUGCUGCAGCGCGCGUAGCAGAAGGCGAGGCUUUUUACGGAGAAUUUGGCCAUCCA
CAAGGGGCCGUCCGCAAUUUCAUUAGCGAGUAAGACAGACUCUUCAGUCUGAUUGAGUGGAAGCAAGCACUAACCUAGAUAGCAUGUGGGUUGCUGACUAUGUGAUCUUA
UUAAAAUUAGGUUAAAUUUCGAGGUUAAAAAUAGUUUUAAUAUUGCUAUAGUCUUAGAGGUCUUGUAUAUUUAUACUUACCACACAAGAUGGACCGGAGCAGCCCUCCAA
UAUCUAGUGUACCCUCGUGCUCGCUCAAACAUUAAGUGGUGUUGUGCGAAAAGAAUCUCACUUCAAGAAAAAGAAGCGCCACUAGUGCAUCGAAUCCUGGGAAACAGGCA
CGGAGAACAAAAUCCGGUAAGCUCGGGAAGCCGUCAACUUCCGCCGUAUUGAUGAAAACAACUAUGAACAACCAACGAAAAAAGACGGCUCGACCGUCUUUCAAUAUGGA
CAGUUGGGGCAUACUUUCUCAUGGACCCCCGAGUUGUGUCUAUUCGUUAACAGCUAGGCCGUCCAUGAGGUCGUAGGCACCCAGUAAUUAAUUGUGGACAGUUCCACUCC
CUUUAGCACACAAAGGUCCAAUCGGGUGCGACUCCCCCCCCUCCUGUGGGCUACAGGAACC
>0.602-0
AUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACAUUCCGAGGGGACCGCCCCCUCGGUAAUGGCGAAUGGGACCCACUGGGAAUAGACUGGGAGAUC
UUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAACAUGUAGCUGGUGGUGAGGAAGAACACAGGAUCUAUGACUAAAAAACCAGGAGCGCCCGGUAAAA
ACCGGGCUAUCAAUAUGGGCCCUGUAGCUCAAUGGUUAGCAGCAGUCCCCUCAUAAGGGAAAGGUUACCAGUUCAAAUCUGGUCUGGGUCAAGCGAGUAAGACAGACUCU
UCAGUCUGAUGUUACGGAGAUGAGUAGAACAAAGUGAGGGACUAUGGGUUAUCAGUCGCCAGUGCGAGGCCCAUUCUCUCUCAAUGGGUAUUGGGUGCAACCCCCCCGUC
CAUCUCGAACGUCAUCGAGACCAGAAAAUGUGUGAUCUGAUUAGAAGUAAGAAAAUUCCUAGUUAUAAUAUUUUUAAUACUGCUACAUUUUUAAGACCCUUAGUUAUUUA
GCUUUACCGCCCAGGAUGGGGUGCAGCGUUCCUGCAAUAUCCAGGGCACCUAGGUGCAGCCUUGUAGUUUUAGUGGACUUUAGGCUAAAGAAUUUCACUAGCAAAUAAUA
AUGCAAUCCGGCCUUUAAACUACCAGCGGGCGGGCUCCCUACGGGGCAGUCCGCCCUUGGCCAACUUAUCGAGUACGCGGGAGUUGUUAGUCUACGUGGACCGACAAGAA
CAGUUUCGAAUCGGAAGCUUGCUUAACGUAGUUCUAACAGGACAGUUGGGGCAUACUUUCUCAUGGACCCCCGAGUUGUGUCUAUUCGUUAACAGCUAGGCCGUCCAUGA
GGUCGUAGGCACCCAGUAAUAUGGGCAAUACGUAAGAGUAGCCCAAAAACUGAAUACACCUACCCAUCAUGGAGAAAGUUCACGUUGACAUCGAGGAAGACAGCCCGUUC
CUUCGAGCUUUACAGCGGAGCUUCCCACAGUUUGAGGUAGAAGCCAAGCAGGUCACCGAUAAUGACCAUGCUAAUGCCAGAGCGUUUUCGCAUCUGGCCUCCAAAUUGAU
CGAGACGGAGGUAGAACCAUCCGAUACGAUCCUAGACAUAGGCAGUGCACCGGCACGCCGAAUGCCCGCGCAAGGAAGGACACGUGAAUGACGUUUCCUUGGGAGUUGAC
GACUCUCCGCCCCCAUUUUUAGGGAAGAGCUGGUCUCCCUCAAAGGGAGACCAGAAACUUCCCCAAUUAUGGUGGCUCAGAUGGGGCCAACUUCGGUUGGGCCGGUUUCU
UCUUGUACCGGUGUUGAGAACCCUGUCUGGGCUACCACCCCUGGUACCCUUGAAAUCACCUCCUAGAUUUCUUCGGAAGGGCUUCGUGAGAAGCUCGUGCACGGUAAUAC
ACUGAUAUUACCAAGAGUGCGGGUAUCGCCUGUGGUUUUCCACAGGUUCUCCAUAAGGAGGGCUCGCGCCCGUAGUCUGGAGAAGCAUCGCCAGGGUUGAGUCGCGGCAG
AACCCGGUUCGCGGACGGCCGCGGCGAGCGGGAAUUGGUCACCCCGCCUAUUUAAAGACCCACAGCCAGCCGACUUCUCCAGUUACGGGAGCGAGCCCCCUUAAAACUGG
AUCUGGGUUGUUCCCACCCAGAUCACCUACAUGGUGUUGUACACUAUUAUUACGGUAAUCUUGUACGCCAGUUUUAUACUCCAGUCGAGACCUGAAGUGGGUUUCCUGAC
GAGGCUGUGGAGAGAGCUUUCGCUUUACUCCCGCACAAGCCGAAACUGGAAGUGUUUCGGCUUCCACUUAAAUCGAAAGCCCCCCAGUCCCCCAGGUUAUGGAGGAAUAA
CCCCGACGUGUUGGAAGGACAAGACUCGGCUUGCUGAGGUGCACACGGCAAGAGGCGAGAAGGACAGUUUCUUACUUUAGAAAAAGUAAGUGGUGCGGAUGGGGUUACCC
CGCCUGGUUUCUACAUUCCAGUAAAGAUGUAGUGGUGGAGUCAACGAACGCGGCCCUAGCCCUGGUAAUGGUGUUAACCAGGGCGAAAGGACUAGAGGUUAGAGGAGACC
CCGCGGUUUGAGAGGUAUCGAUGCUGGAUCCUGGGAAACAGGCUUGACAGGUUUGGGUGGUUCCAAACCGCUGCAUCACUCGAGAUACUCAGCUCAUCGCCCAGGGAGGG

AGAGCCAAGAUCUGUGGCAUAUAUCUCUUCAACUGGGCGGUGAGAAUGCAGGUUUCCCCAACUGACACAAACCGUGCAACUUGAAGCUCCGCCUGGUCUUUCCAGGUCUA
GAGGGGCGACACUUUGUACUGUGUUUGGCUCCACGCUUGACUCAUCGGCGAGUGUUAGUAGCAGCACUGUUGCUUCGUAGCGGAGCAUGAUGGCCGUGGGAAUUCCUCCU
UGGUAACAAGGACCCACGGGGCCGAAAGCCACGUCCACACGGACCCAUCAUGUGUGCAACCCCAGCACAGCAACUUUACUGUGAAACUCACUUUAAGAUGACGUUGAUAC
UGGUACCCAAGCACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUUUAAAAGCGCCCGGU
UUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUCCUUUACAACUAUUGACAUACGACGAACUGUGGCAGCACACCAUUUGGUGACGGGAAGCUGGU
CGCUCCCGACGCAGGUCGUCGUAAGAACUUUGUGAGACCAAAACCGUGUCAGAUCGCGAAAGCGCCACUUCGCCGAGGAGUGCAAUCUGUGAGGCCCCAGGAGGACUGGG
UACACCACUAGCACAAAUCGGAUCCUGGGAAACAGGCAGAACUACGGUUCAUAAGCUCGGGUAGGCCGUCAACCUACCGCCGUAUCGUAUUGUGUUUGGCCCAGUAGUUC
GCCUGUGUGAGCUGACAAACUUAGUAGUGUUUGUGAGGAUUAACAACAAUUAACACAGUGCGAGCUGUUUCUUAGCACGAAGAUCUCGAUGUCUAAGAAACCAGGAGGGC
CCGGCAAGAGCCGGGCUGUCAAUAUGCUAAAAC
>0.602-1
UCGGGUCGGCAUGGCAUCUCCACCUCCCCGUGGUCCGACCUGGGCAUCCGAAGGAGGACAGACGUCCACUCGGAUGGCUAAGGGAGUGCCAGCCCCUCAAUAGAGGGGGG
GCGGUUCUUGUUCUCCCUGAGCCACCAUCACCCAGACACAGAUAGUCUGAGAAGGAGGUGAUGUGUGACUCGGAAAAACACCCGCUAUGAACAACCAACGGAAAAAGACG
GCUCGACCGUCUUUCAAUAUGGACCUCGUGGCGCAACGGUAGCGCGUCUGACUCCAGAUCAGAAGGCUGCGUGUUCGAAUCACGUCGGGGUCAAGCUAGUAGGACAGACU
CUUCAGUCUAGGUUGGUGGAAGCAGGAGAACACUGCUUGACUAGCGGAGAUAAUUGUGGACAGUUCCACUCCCUUUAGCACACAAAGGUCCAAUCGGGUGCGACUCCCCC
CCCUCCUGUGGGCUACAGGAACCGCAAAAAUGUGAUCUUGCUUGUAAAUACAAUUUUGAGAGGUUAAUAAAUUACAAGUAGUGCUAUUUUUGUAUUUAGGUUAGCUAUUU
AGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGGAAGCCCUCUCUGCGGUUUUUCAGAUUAGGUAGUCGAAAAACCUAAGAAAUUUACCUGCUACAUU
UCAAGUUGCGGAGCCUUUAAACUACCAGCGGGCGGCAGGGUCAAGGCCCUUGCCGCCCUUUCUCAACUUGCAGGCUACGACUUAGUUGUUGAUCUGUGUGAAUCAGACUG
CGACAGUUCGAGUUUGAAGCGAAAGCUAGCAACAGUAUCAACAGCCAGGCUUGGGCGUAUUUUCUCCGGUGAGCGUAGCUGCCUGUAGUCGUUAGCAGCUACGUUGUCCG
CGGGAUUCUAGCCACCUGGCAAUUUCGCCUUUCUUGGAUCCCUUGUUACUCCACCGCAGUGAGAGAUAAACAACCCAGAAUGAAGGUCACUGUAGAUGUUGAGGCUGACA
GCCCAUUUUUGAAGGCCCUUCAGAAAGCAUUCCCGGCUUUUGAGGUUGAAUCACAGCAGGUCACACCUAAUGACCAUGCUAAUGCCAGAGCUUUUUCGCAUCUGGCCACA
AAGUUAAUUGAACAAGAGGUUCCAGCCAACAUCACCAUCCUGGACGUGGGCAGUGCGCCCGCAAGGAGGUUGUCCACGCAAUGGGAGGCAUCAUUUCGCCUCCGGGUGCU
GACUACUCCCCGUCCCCGUUUUUACGGAGAAUUUGGCCAUCCACAAGGGGCCGUCCGCAAUUUCAUUAGUUAUGGUGGGCUGGGCGGGGGCAUCGAAAGAUGCGCCGGUU
UCCUUGAUCACCGGUUACGCCAACCCCGUUCAGUUCACCACCAGCGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUG
CAUAGCAAGUCUUAGAAUGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGGACUCCCCUCUGUGGCCUGGCGGAAAAGUCACAAGGGUACCACAGC
GAGGAACCCCGGUUCGAAACCGGCAGGAUCCGCUGUGAGCGUAUAUAGGGCGUCUGCGCGUAGAACCCAGCCAAAGACCCCCAGACACGGAGAGGAGUCUUUCACCCACC
CAUAGGGCCCACCGGGCGCCAGCACUCUGGUAUCAUGGUACCUUUGUGCGCCUGUUUUAUCAUAAGUCUGGGCUUAGCCACUGACGAGUCGCUGGGAUACGGCGAAACUU
AUGAGUGCUUUCCAGUUCACUUAAAUAGAACUGGCCCCAGUUCCUAGGUCCAGGUAGGAUCCCGUGGCGCUGUGAGGGCGGGACUCGGUCUGCUGCAGCGCGCGUAGCAG
AAGGCGAGGCCCGAGUCUUGGAAUGACUUAAAUCUUACCCUGGUGCGGAUGGGAUCUCUCUCCCGCCUGGUUUCUUGCUUCCAGUCAAAAAGCAAGUGGUGGUGCCAAAU
GGAGGCCAUGCGCCACGGAAGCUGUACGCGUGGUAUAUUGGACUAGCGGUUAGAGGAGACCCCUCCCAUCACUUGACGUACAAGCCGGAUCCUGGGAAACAGGUUUAACG
GGCUCACUGUGGUGGUGGGCCGUCGAUACACUUGUAUGUGCCCCAACCUCAUCAGCCGGGGGGGGAGGGCGGCCACGUGCGGUCGCUACCUCUUCAACUGGGCUGUGAAG
CUAGCCGUCGCGUCUUAACGCGCAGCCCUGAAGGCCCCACACCUUGUGGAUCUUGCCGUGGGUAUGUUUCUGGCAUGUGUUUCUCAAGCCUGCAACCGAAGCCGAACAGC
CACAUGAACAGUUUGAGCGUGGUAGCGCUGUGUGAGUUGGCGGUGGAUCCCCCUCGUGGUAACACGAGCCCCCGUGGCCAAAAGCCCAGUGUUUACAGCACCUCUCACAU
CCAGGACGACCCCAUCCUGGCGCUCACUCUUAGUAGUAUGGCUUAGUACGCAUUAGGUGGUAAGCCGAGCUCUCCCUCGGCCUUGUUCUGAAUGCACACAUGUCUAGGGG
CUAAGGAUGUCCUACAGGUACCCGCACGUAACCUUCAGAGAGUGCGGAUCUGAGUAGGAGACCGUGGUGCACUGCUUUACAGAUGCAGCCCGGUUUAAAAAGCGUCUAUG
CCCCUACAGGGUAGCGGUGGGCCGCGCCCUUUCCUUUUAAAACUACUUGUUCAGGGGGUGAUGUGGCAGCGCACCACGACAUCGUGACGGGAAGAGGUCGUCCCCCACGC
AUCAUCUCUCUAGGGCAUUUUCGUGAGACCGAAAGUCAGGCCGGGAAGUUCCCGCCACCGGAAGUUGAGUAGACGGUGCUGCCUGCGACUCAACCCCAAGGAGGACUGGG
UUGGUGAGUUGCUGUCGUGGGAUCCUGGGAAACAGGUUCGGUGAACAAAAAGCCGGUAAUCUCGGGCGUUCGUCAGACGCCGUCGUAUCAACACAGUAUGCAUAGAAGUU
CAUCUGUGUGAACUUUGUGAUUGACAGCUCAACACAAGUGCGGACUACCGUAAACACAGUUUGAACGUUUUUUGGAGAGAGACUACUAUGUCUAACAAAAAACCAGGAAA
ACCCGGGGCAACCCGGGUUGUCAAUAUGCUAAAGC
>0.652-0
AUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACAUUCCGAAGGGACCGUCCCUUCGGUAAUGGCGAAUGGGACCCAUUGACUAUGUGAUCUUGCUUU
CGUAAUAAAAUUCUGUACAUAAAAGUCGAAAGUAUUGCUAUAGUUAAGGUUGCGCUUGCCUAUUUAGGCAUACUUCUCAGGAUGGCGCGUUGCAGUCCAACAAGAUCCAG
GGACUGUACAGAAUUUUCCUAUACCUCGAGUCGGGUUUGGAAUCUAAGGUUGACUCGCUGUAAAUAAUAGUCGUUCGUCUGUGUGAGCUCUACUACUUAGUAUUGUUUUU
GGAGGAUCGUGAGAUUAACACAGUGCCGGCAGUUUCUUUGAGCGUUGAUUUUCAAUGUCUAAGAAACCAGGAGGGCCCGGAAGAAGCCGGGCCAUCAAUAUGCUGAAACA
UGAUUAAAAAACCAGGAGGGCCCGGUAAAAACCGGGCCAUCAAUAUGGGGUGGUGUGUGCCACCCCUGAUGAGUCCGAAAGGACGAAUCAGUCGCCAGUGCGAGGCCCAU
UCUCUCUCAAUGGGUAUUGGGUGCAACCCCCCCGUCCAUCUCGAACGUCAUCGAGACCAAAGCAGGUCCCCCCAACUGACACAAAUCGUGCAACUUGAAACUCCGCCUGG
UCUUUCCAGGUCUAGAGGGGUGACACUUUGUACUGUGUUUGACUCCACGCUCGGUCCACUGGCGAGUGUUAGUAACGGCACUGUUGCUUCGUAGCGGAGCAUGACGGCCG
CGGGAAUUCCUCCUUGGUAACAAGGACCCGCGGGGCCAAAAGCCACGUCCCAACGGACCCGUCAUGUGUGCAACCCCAGCACAGCAACUUUUCUGUGAAACCCACUUCAA
GGUGACACUGAUACUGGUACUCAAACACUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGCAACACGCGACACUCGGGAUCUGAGAAGGGGACUGGGGGUUCUA
UAAAACCGCCCAGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUCCUUUAUUACCACUGAAUUCCACGGCCCAAGUCUCGUCCAGGAUGCAAU
GGACGAGAUGUAAGGACUAGAGGUUAGAGGAGACCCCGUGGAAACAAAUUAUGGUGGCUCAGGCAGGGCAGCUUUCGGGCUGGCCGGUAGCUGAUUGGGCCGGUAUUCCU
ACCCUUGUCUGGGUCACCACCAACCCCCAGUCCCCCAGGUUAUGGAGGAAUAACCCCGACGUGCUGAAAGGGCAGCUGCCCUUGGGUUUUACUCCUUGAACCCUUCGGAA
GAACUCUUUGGAGUUCGUACCAGUACCUCACAUAGUGAGGUAAUAAGACUGGUGGGCAGCGCCUAGUCGAAAGACUAGGUGAUCUCUAAGGAGGGCUCGCGCCCGUAGUC
UGGAGAAUCAGUCGCCAGGGUUGCGUUGCGGUAUGCCCCGGUUGGAGCCUAAGCGCGGCUCGUAUCGGCCGGUUUCCGCGACAAGCAGGGUUUGGCAGCCCCGUUAUUUC
CAAGACCCCGCCAGCCGACUUCUCCAGUUUACGGGAGCGAGCCCUUAGCUCAUUUCCCAGGGAGGGAAGGCUAAAGUCUGCGGCCUUUACCUCUUUAACUGGGCAGUGCG
CCUGGGAAUAGACUGGGAGAUCUUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUAGCUGGUGGUGAGGAAGAACACAGGAUCUACACCACUA
GCACAAAUCGGAUCCUGGGAAACAGGCAGAACUACGGUUCAUAAGCUCGGGUAGGCCGUCAACCUACCGCCGUAUCGUAUUGUGUUUGGCCCUCCGUGUAGCUCAGUUUG
GUAGAGCGCCUGAUUUGGGAUCAGGAGGUCCAAGGUUCAAAUCCUUGUAUGGAGAGCCCCCGCAAGGAGGAACGUGUCACUGGCAUGUUCCUGGGAGUUAACGGCUCUCC
GGCCCCACACAGUCAGGCCGAAAGCCACCACAAGCGGUACAGUGGUGCUGCCUGCAUUAACCGCCCCAGGUGGACUGGGA
>0.652-1
UCGGGUCGGCAUGGCAUCUCCACCUCCCCGCGGUCCGACCUGGGCAUCCGAAGGAGGACAGAACGUCCACUCGGAUGGCUAAGGGAGAGCCAGUCCAACAAUGUGAUCUU
GCUUGCGGAGGCAAAAUUUGCACAGUAUAAAAUCUGCAAGUAGUGCUAUUGUUGGAAUCACCGUACCUAUUUAGGUUUACGCUCCAAGAUCGGUGGAUAGCAGCCCUAUC

AAUAUCUAGGAGAACUGUGCUAUGUUUAGAAGAUUAGGUAGUCUCUAAACAGAACAAUUUACCUGCUGAACAAAUUAGAAGUUCAUCUGUGUGAACUUAUUCCAAACAGC
UUUUGGAGUAGUGCGUGUGAACGUAAACACAGUUUGAACGUUUUUUGGAUAGAGACAACUAUGUCUAACAAAAAACCAGGAAGACCCGGCUCAGGCCGGGUUGUCAAUAU
GCUAAAGCAUGAACAACCAACGGAAGAAGACGGGUCGACCGUCUUUCAAUAUGCAUAAGUCUGGGCUUAGCCCACUGAUGAGGCGUUGAGAUACGGCGAAACUUAUGGAA
GAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGACCCUAGCCGUCGCGUCUUAACGCGCAGCCCU
GAAGGCCCCACACCUUGUGGAUCUUGCCGUGGGUAUGUUUCUGGCAUGUGUUUCUCAAGCCUGCAACCGAAGCCGAACAGCCACAUGAACAGUUUGAGCGUGGUAGCGCU
GUGUGAGUUGGCGGUGGAUCCCCCUCGUGGUAACACGAGCCCCCGUGGCCAAAAGCCCAGUGUUUACAGCACCUCUCACAUCCAGGACGACCCCAUCCUGGCGCUCACUC
UUAGUAGUAUGGCUUAGUACGCAUUAGGUGGUAAGCCGAGCUCUCCCUCGGCCUUGUUCUGAAUGCACACAUGUCUAGGGGCUAAGGAUGUCCUACAGGUACCCGCACGU
AACCUUCAGAGAGUGCGGAUCUGAGUAGGAGACCGUGGUGCACUGCUUUACAGAUGCAGCCCGGUUUAAAAAGCGUCUAUGCCCCUACAGGGUAGCGGUGGGCCGCGCCC
UUUCCUUUUAAAACUACUUGUUCCCGGGCCCAACCCAGCUAGGGAUAGCCUUUCUGGGGUAAGGACUAGAGGUUAGUGGAGACCCCCGGCUUUUGUAUUAUGGUGGGCUG
UGUGGGGGCACCGAAAGGUGCGCCGGGUCCUUUUAGCCGGUUACGCCAACCCUGCACAGUUCACCACCAACCCCCAGUUCCUAGAUCUAGGAAAGAUCCCGUGGCGUUGU
GAGGACGGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAAGUCUUAGAAUGCGGGUACCGUACAGUGUUG
AAAAACACUGUAAAUCUCUAAAAGAGGACUCCCCUCUGUGGCCUGGCGGAAAAGUCACAAGGGUACCACAGCGAGGAACCCCGGUUCGAAACCGGCAGGAUCCGCUGUGA
GCGUAUAUAGGGCGUCUGCGCGUAGAACCCAGCCAAAGACCCCCAGACACGGAGAGGAGUCUUUAAGCUACUGUCCCAAGGGGGGAGGGCUGCCACUUGUGAAUACCUCU
UCAAUUGGGCGGUAAAGCCCAGGAAGCUGGGGGGGCGGUUCUUGUUCUCCCUGAUCCACCACCAUCCAGGCACAGACAGCCUGACAAGGAGAUGGUGAUUUGGUGAGUUG
CUGUCGUGGGAUCCUGGGAAACAGGUUCGGUGAACAAAAAGCCGGUAAUCUCGGGCGUUUGUCAGACGCCGUCGCAUCAACACAGUAUGCAUGGAGGCGUGGCAGAGUGG
UUUAAUGCACCGGUCUUGAAAACCGGCAGUCGCUCCGGCGACUCAUAGGUUCAAAUCCUAUCGCCUCCGGGCAGCGCAAGGAAGGUGGGCUGGAUGCCGCCUUAGGUGGU
GACGACACCUCGCCCCCAAGAUGUCAGGCCGGAAACGCCACCGGAUGGUCGUAAACGGUGCGGCCUGUGACGACCCCCGGGCGGACCGGGU
>0.699-0
AGGGAUAGACCGGAGAAAUACACAAUUUCUGGGCACUAAACAACUACAAGCACAGAGCGCUUCGAGAAGUAGUUGUUUCUUGACGAGAGAAUAUGUAUAUCCCAACAAUG
UGAUCUUGCUUGCGGAGGCAAAAUUUGCACAGUAUAAAAUCUGCAAGUAGUGCUAUUGUUGGAAUCACCGUACCUAUUUAGGUUUACGCUCCAAGAUCGGUGGAUAGCAG
CCCUAUCAAUAUCUAGGAGAACUGUGCUAUGUUUAGAAGAUUAGGUAGUCUCUAAACAGAACAAUUUACCUGCUGAACAAAUUAGCUUAUCGCCCAAGGAGGAAAGGCCA
AGAUUUGUGGCCUCUACCUUUUCAAUUGGGCGGUACGCACACCACUAGCACAAUCCGGAUCCUGGGAAACAGGCAGAACCUCGGUUCAUAAGCUCGGGUAGGCCGUCAAU
CUACCGCCGUAUCGUUUUGUGUUUGGCCGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAAGUCUUAGAA
UGCGGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAGAGCUAGUAGGACAGACUCUUCAGUCUAGGUUGGUGGAAGCAGGAGAACACUGCUUGACUAGCG
GAGAAGGGGCACACUGGCAACACACCAUUGGUGACGGGAAUUAGGUCGACCCCGCCAGUGGGCUCCUCAAAGACAGAAUGUGAGACCAAUUUAUGGCGGGCUGUGUGGGG
GAGCCGAAAGGCUCGCCGGAUGUUUCGACCGGUAGUGCUAACCCCGCACAGUUCGCCACCACGGGAGGCCAUGCGCCACGGAAGCUGUACGCGUGGCAUAUUGGACUAGC
GGUUAGAGGAGACCCCUCCCAUGACAAGCAGGCCUCCCCAACUGACACAAACCGUGCAAUUUGGAACUCCGCCUGGUCUUUCCUGGUCUAGAGGGGUGACACUUUGUACU
GUGUUUGGCUCCACGCUCGGUCCACUGGCGAGUGUUAGUAACAGCACCGUUGCUUCGUAGCGGAGCAUGAUGGCCGUGGGAACUCCUCCUUGGUAACAAGGACCCACGGG
GCCGAAAGCCACGUCCAAUCGGACCCAUCAUGUGUGCAACCCCAGCACAGCAACUUUUCUGCGAAACUCACUUCAAGGUGACACUGAUACUGGUACUCAAACACUGGUGA
CAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGCGUCACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUAUAAAAGCGUCCAGGUUAAAAAGCUUCUAUGCCUG
AAUAGGUGACCGGAGGCCGGCACCUUUUCUUUACAGCCACUGACUUGUGGCCGGCAUGGUCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACAUUCCGAGGGGACCGUCC
CCCGGAAUGGCAGAUGGGACCCACCUCGUCAGAGGGGGGGCGGUUCUUGUUCUCCCUGAGCCACCAUCACCUAGACACAGAUAGUCUGAAAAGGAGGUGAUGCGUGUCUC
GGAAAAACACCCGCUGAACUACAGUUCCGCUCCUCUAGCACACAGAGGUCAAUUGGGUGCGACCCCCCCCUCUCCCGUGGGUCAACGGGAACCGCCCACGCAAGGGGAGG
CAUCAUUUCGCCUCCGGGUGCUGACGACACCCCGCCCCAGGGGUGGUGUGUGCCACCCCUGAUGAGUCCGAAAGGACGAAGGCUCGCGCCCGUAGUCUGGACAAGCAUCA
CUAGGGUUAAGUUACAGCAGAACCCGGUUCGCGGACGGCCGCGGCAAGCGGGACUUAGUUACCCCGCCAAUUUAAAGACCCGCAGCCAGCCGACUUUUCCAGUUACGGGA
GCGAGCCCCCGAAGGUGAGCCUUUAAACUACCAGCGGGCGGACCUCCUACGGGAGAGUCCGCCCUUGGAAGGCUUAUUCAGCUUGGGAAAUGUCUAAGAAACCAGGAGGG
CCCGGAAGAAACCGGGCCAUCAAUAUGGCUGAUUUAGCUCAGUAGGUAGAGCACCUCACUUGUAAUGAGGAUGUCGGCGGUUCGAUUCCGUCAAUCAGCA
>0.699-1
UAGGAAAGACUGGAGAUACCUUUGGUUCUUCGACAUUAACUCCAAUCGGCACAGAGCGCCGUGGGAAACGGAGUUAAUAUGGAAAAAACAUAAGUAUCCUGACUAUGUGA
UCUUAUUAAAAAUUAGGUUAAAUUUCGAGGUUAAAAAUAGUUUUAAUAUUGCUAUAGUCUUAGAGGUCUUGUAUAUUUAUACUUACCACACAAGAUGGACCGGAGCAGCCC
UCCAAUAUCUAGUGUACCCUCGUGCUCGCUCAAACAUUAAGUGGUGUUGUGCGAAAAGAAUCUCACUUCAAGAAAAAGAACCUCAUCAGCCGGGGGGGGAGGGCGGCCAC
GUGCGGUCGCUACCUCUUCAACUGGGCUGUGAAGUGGUGAGUUGCUGUCGUGGGAUCCUGGGAAACAGGUUCGGUGAACAAAAAGCCGGUAAUCUCGGGCGUUUGUCAGA
CGCCGUCGCAUCAACACAGUAUGCAUGGUGCCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACUUCGGUCCGUGUACUUCUAGCACAAGAUGCUAGUUUCAGGGUAC
GGGCACCCCCCCCUCAUUUAAUGGGGGGCCUCCUCAAGGAGACACCUGACUCCAGACUCUUCAGUCUGAGUUUGUGGAGAUGAGUGUAAAUCUGGCAUAGCAUACAGGUU
AAGGGGAUGGUGUGGCAGCGCACAACGACAUCGUGACGGGAGUGGGUCGCCCCCGACGCACCAUCCUCUUGGGAAAAAUUUUCGUGAGACCAAUUAUGGUGGGGCGUGAU
GGGGAGGCUUCGGCCUGCUGGUUUCACUAACGCCAGUCUGUCAACCCUGUCACGUCCUGCCACCUGUCCGGGCCCAACCCAGCUAGGGAUAGCCUUUCUGGGGUAAGGAC
UAGAGGUUAGUGGAGACCCCCGGCUUUUGCUAGCCGUUGCGCCUUAGCGCACAGUCCUGAAGGCCCCACAUCUUGUGGAUCUUGCCGUGGACAUGUUUCUGGCAUGUGUU
UCUCAAGCCUGCAACCGACGACUGAACAGACGUGUGAACAGUUUGAGUGUGGUAGCGCUGUGUGAGUUGGCGGUGGAACCCCCUCGUGGUGACACGAGCCCCCGUGGCCA
AAAGCCCAGUGUUUACAGCACCUCUCACAUCCAGGGAAACCCCACCCUGGCGCUCACUCUUAGUAACACUUCCUAGUAGGUAUUAGGUGGUAAGCCGAGCUCUUCCUCGG
UCUUGUUCUGAAUGCACACAUGUCUAGGGGCUAAGGAUGUCCUACAGGUACCCGCACGUAACCUUCAGAGAGUGUGGAUCUGAGUAGGAGACUGUGGUGCGCUGCUUUAC
AGACGUAGCCCAGUUUAAAAAACGUCUAUGCCCCUACAGGGUCGCGGUGGGCCGCACCCUUUCCUUUUACAAACUUGGUUUUUCGGGUCGGCAUGGCAUCUCCACCUCCC
CGCGGUCCGACCUGGGCAUCCGUAGAGAGACGAACGUCCACUCGGAUGGCUAAGGGAGAGCCAGUCUGGGAUAGACUAGGAGAUCUUCUGCUCUGCACAACCAGCCACAC
GGCACAGUGCGCCGACAAUGGUGGCUGGUGGUGCGAGAACAGGAUGAGACGAGUUAGUCUCGACCAUUUUCACACAAUGGUUCUGAGGGGUGCAACUCCCCCCCUUAACU
AGAGGGAAAUCUAGCCCAACCCCGUAAGGAAGGACAAGGCUGUCCUUGAGUGCCAACGACGCUCCGGCCCCAGAUAAGUCUGUGCUUAGCACACUGAUGAGUCUCUGAAA
UGAGACGAAACUUAUCGACUCCCUUCCGUGGCUUGGUGGAAAAGUCACAAGGGUACCAUAGCGAGGAACCCCGGUUCGAAACCGGCAGGAUCCGCUAUGAGCGUAAGUGA
GGCGCUUGCGCGUUGAACCCGGCCAAAGACCCCCAGACACGGAAAGGAGUCUUUGUUGCGGAGCCUUUAAACUACCAGCGGGCGGCAGGGUCAAGGCCCUUGCCGCCCUU
UCUCAACUUGCAGGCUACGACUUAUGAACAACCAACGGAAGAAGACGGGUCGACCGUCUUUCAAUAUGGGAGGCGUGGCAGAGUGGUUUAAUGCACCGGUCUUGAAAACC
GGCAGUCGCUCCGGCGACUCAUAGGUUCAAAUCCUAUCGCCUCCG
>0.743-0
UCCAGUCGAGACCUGAAGUGGGUUUACUGAUGAGGCUGUGGAGAGAGCGAAAGCUUUACUCCCACACAAGCCGAAACUGGAGCGAGAAUGGUCAAAUUGGUAAAGGCACA
GCACUUAAAAUGCUGCGGAAUGAUUUCCUUGUGGGUUCGAGUCCCACUUCUCGCAGCUCCCGUAAGGAAAGCGCAAGCUUUGAGCAUUGACAACGCUCCGGCCCCAACGC
AGGAAUCUUAGACCAACCCAAACCCGUGCAACUGCAAGUUUCGCCCGGUCUUUCCGGGUCUAGAGAGACAAACAGAUGUACUGAGACUGACUCCACGACUGGUCUACUGG

CGGGUGCUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGUGGGAACACCCCCGUGGUAACACGGACCCACCGGGCCAAAAGCCACGCCUCACGGCCUCAUG
UGUGUGCAACCCCAGCACGGCAAAUUGUUUGUGAAACACAUCUUAAGGUAACACUGGGACUGGUACUUAGUUUCUGGAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGA
GGUAACAAGAGACACUCGGGAUCUGAGAAGGGGACUGGAAGUUCUUUAAAACUGUCCAGUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCC
UUUUAUCCAACACAUUUAAUUAUGGUGGCUCAGAUGGGGCCAACUUCGGUUGGGCCGGUUUCUUCUUGUACCGGUGUUGAGAACCCUGUCUGGGCUACCACCCCUUGGGA
AAGACCAGAGGUACUCGUUGAUUCACCGCCACCAAACUACACGGCACAGCACGCCGAAAAAGGUAGUUUGGAGGUUGUAAAACAGCAAGUAACUGGCAUGAAAAUGUCGG
GCCUGACGAACCCGCGCACCCGAGUCCCCCAGUUGGGGAAGAUGAAUGACCAACGGAAAAAGGCGAGAAACACGCCUUUCAAUAUGGUGGCCGGCAUGGCCCCAGCCUCC
UCGCUGGCGCCGGCUGGGCAACGAUCCGAGGGAGCUACUCCUCUCGAGAAUCGGCAAAUGGGGCCCCGAAAAUGUGUGAUCUGAUUAGAAGUAAGAAAAUUCCUAGUUAU
AAUAUUUUUAAUACUGCUACAUUUUUAAGACCCUUAGUUAUUUAGCUUUACCGCCCAGGAUGGGGUGCAGCGUUCCUGCAAUAUCCAGGGCACCUAGGUGCAGCCUUGUA
GUUUUAGUGGACUUUAGGCUAAAGAAUUUCACUAGCAAAUAAUAAUCUGGGAGAGACCAGAGAUCUGCUGUCUCUACAGCAUCAUUCCAGGCACAGAACGCCAGAAAAUG
GAAUGGUGCUGUUGAAUCAACGGAUCUGAAGAGCGGCCAGUUGCUGCUGCGAUCAAGACACGAUCGUUCAAAAGGGUGCAACUCCCCCCCCCCUUGGAGGGUAUCCAAGAC
CUGGGGGCCCAAGGUGAGAUGAAGCUGUAGUCUCACUGGAAGGACUAGAGGUUAGAGGAGACCCCCCCAAAAAAGACUCCUCUCCGUAGUCUGGAGGUUAGGUCGCAAGG
GUGCGGCGGCGGGGAACCCUGGUUCGAAACCAGCUGGAUCCGCCGUCCCGAUGCGCUUGGCCCCGCAUCCACGACGGCCCCGGGCGUCGAGACCCAGCCGCGAUACGCAC
ACCCCAAAUACGGAGGGGAGUCUUUGGUGCCUUUGAGAGUUACUCUUUGCUCUCUUCGGAAGAACCCUUAGGGGUUCGUGCAUGGGCUUGCAUAGCAAGUCUUAGAAUGC
GGGUACCGUACAGUGUUGAAAAACACUGUAAAUCUCUAAAAGAG
>0.743-1
CCGGUGUCUCAAGGUGCGUACCUUGACUGAUGAGUCCGAAAGGACGAAACACCAGGGGUCGUUAGCUCAGUCGGUAGAGCAGCGGACUUUUAAUCCGUUGGUCGAAGGUU
CGAAUCCUUCACGACCCAUUCAGCGCAAGGAAGGGAAGCUUGAGGCUACCUUAGGUGGUGACGACACCUCGCCCCCACUAGCCGUCGCGUCUUAACGCGCAGCCCUGAAG
GCCCCACACCUUGUGGAUCUUGCCGUGGGUAUGUUUCUGGCAUGUGUUUCUCAAGCCUGCAACCGAAGCCGAACAGCCACAUGAACAGUUUGAGCGUGGUAGCGCUGUGU
GAGUUGGCGGUGGAUCCCCCUCGUGGUAACACGAGCCCCCGUGGCCAAAAGCCCAGUGUUUACAGCACCUCUCACAUCCAGGACGACCCCAUCCUGGCGCUCACUCUUAG
UAGUAUGGCUUAGUACGCAUUAGGUGGUAAGCCGAGCUCUCCCCUCGGCCUUGUUCUGAAUGCACACAUGUCUAGGGGCUAAGGAUGUCCUACAGGUACCCGCACGUAACC
UUCAGAGAGUGCGGAUCUGAGUAGGAGACCGUGGUGCACUGCUUUACAGAUGCAGCCCGGUUUAAAAAGCGUCUAUGCCCCUACAGGGUAGCGGUGGGCCGCGCCCUUUC
CUUUUAAAACUACUUGUUCAUCAAUGGUGGGCUGGAUGGGGGAGCCGAAAGGCUCGCUGGGUCCUUGUUAGCCGGUAGUCUCAACCCUGUUCAGUUCACCGCCAAUUAGG
AAAGACUGGAGAUACCUUUGGUUCUUCGACAUUAACUCCAAUCGGCACAGAGCGCCGUGGGAAACGGAGUUAAUAUGGAAAAAACAUAAGUAUCGGCACAUUAGUGUCGG
GCGUGACGCACCCGCUCCCCUCAGUCCCCUGUGCAACAGACUAAAAAACCAGGAGGGCCCGGUAAAAACCGGGCUAUCAAUAUGUCGGGUCGGCAUGGCAUCUCCACCUC
CUCGCGGUCCGACCUGGGCAUCCGAAGGAGGACGCACGUCCACUCGGAUGGCUAAGGGAGAGCCACUGUUAAGAUGUGAUCUUGCUUCCUUAUACAAUUUUGAGAGGUUA
AUAAGAAGGAAGUAGUGCUAUCUUAAUAAUUAGGUUAACUAUUUAGUUUUACUGUUCAGGAUGCCUAUUGGCAGCCCCAUAAUAUCCAGGACACCCUCUCUGCUUCUUAU
AUGAUUAGGUUGUCAUUUAGAAUAAGAAAAUAACCUGCUAACUUUCAACCCUCAACAGAGGGGGGGCGGUUCUUGUUCUCCCUGAGCCACCAUCACCCAGACACAGAUAG
UCUGACAAGGAGGUGAUGUGUGACUCGGAAAAACACCCGCUUUAGCUCGCCAGUGCGAGGCUCGUUCCCACACAACAAGUAAUGGUGGUGCAACUCCCCCGUCCUCCCGA
ACGUCAUCGGGACCACCGGGCCCAACCCAGCUAGGGAUAGCCUUUCUGGGGUAAGGACUAGAGGUUAGUGGAGACCCCCGGCUUUUGGGCUCGCGCCCGUAGUCUGGACA
AGCAUCACUAGGGUUAAGUUACAGCAGAACCCGGUUCGCGGACGGCCGCGGCAAGCGGGACUUAGUUACCCCGCCAAUUUAAAGACCCGCAGCCAGCCGACUUUUCCAGU
UACGGGAGCGAGCCCCCGGCACCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACUUCGGUCCGUGUACUUCUAGCACAACGUGCUAGUUUCAGGGUACGGGUGCCCC
CCCAUUUUCGUGGGGGCCUCCAAAAGGAG
>0.798-0
GCCCCCGCAAGGAGGAACGUGUCACUGGCAUGUUCCUGGGAGUUAACGGCUCUCCGGCCCCAGACCACGUUCCUUUAAACUGCCAGGGGGCGGGCUCGAGCCUUCGGGCG
AGUCCGCCCUUGGACGGCUUAUCCAACUCGGAGAGUGGCCGGCAUGGCCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACGAUCCGAGGGAGCUACUCCUCUCGAGAAUC
GGCAAAUGGGGCCCCCCGGGCCCAACCCAGCUAGGGAUAGCCUUUCUGGGGUAAGGACUAGAGGUUAGUGGAGACCCCCGGCUUUUGAUGAAUAACCAACGGAAAAAGGC
GAAAACCACGCCUUUCAAUAUGACGCAGGAAUCUUAGACCAACCCAACCCGUGCAGCUGCAAGUUUCGCCCGGUCUUUCCGGGUCUAGAGAGACAAACAGAUGUACUGAG
AUCAACUCCACGAUUGGUCUACUAGCGGGUACUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGUGGGGAACUCCCCCAUGGUAACAUGGACCCACCGGGCC
AAAAGCCACGCCUCACGGCCUCAUGUGUGUGCAACCCCAGCACGGCAAUUUGUUUGUGAAACACACCUUAAGGUAACACUGAGACUGGUACUUGGUUUCUGGAGACAGGC
UAAGGAUGCCCUUCAGGUACCCCGAGGUAACAAGAGACACUCGGGAUCUGAGAAGGGGACCAGGAGUUCUAUAAAACUGCCUGGUUUAAAAAGCUUCUAUGCCUGAAUAG
GUGACCGGAGGCCGGCACCUUUUCCUUUACUCAAAUUUCAUUGGCCCUGUAGCUCAAUGGUUAGCAGCAGUCCCCUCAUAAGGGAAAGGUUACCAGUUCAAAUCUGGUCU
GGGUCAUCCAGUCGAGACCUGAAGUGGGUUUACUGAUGAGGCUGUGGAGAGAGCGAAAGCUUUACUCCCACACAAGCCGAAACUGGACAAUGUCAGACCACGCUACGGAC
GUGGCUACGUCUAGGCGGAGAGUGGCAGGUCUGCGAUAGUGCCCCAGGAGGACUGGGUAAUUAUGGUGGGGCGUACAGGGGCAUCGCAAGAUGCGCCGGUAAGGGUAGCC
GCCGGUAACGCCAACUCUGUACGUCUCACCACCUCUGCAAAAAUGUGAUCUUGCUUGUAAAUACAAUUUUGAGAGGUUAAUAAAUUACAAGUAGUGCUAUUUUUGUAUUU
AGGUUAGCUAUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGGAAGCCCUCUCUGCGGUUUUUCAGAUUAGGUAGUCGAAAAACCUAAGAAAUU
UACCUGCUACAUUUCAACUGGGAAUAGACUGGGAGAUCUUCUGCUCUAUCUCAACAUCAGCUACUAGGCACAGAGCGCCGAAGUAUGUAGCUGGUGGUGAGUAAGAACAC
AGGAUCUGUUGCCUUUGAGAGUUACUCCUUGCUCUCUUCAGAAGUUCUCUACCGAGAACGUGCAGUGGCUGUCAUAGACAGCCUUCAGACUGCGGGCAACGUCAGUUCCA
UAAGGAACUGAUAAUCUCCAAAAGGAG
>0.798-1
UUCAGCGCAAGGAAGGGAAGCUUGAGGCUACCUUAGGUGGUGACGACACCUCGCCCCCAGUUGCGGAGCCUUUAAACUACCAGCGGGCGGCAGGGUCAAGGCCCUUGCCG
CCCUUUCUCAACUUGCAGGCUACGACUUUCGGGUCGGCAUGGCAUCUCCACCUCCCCGCGGUCCGACCUGGGCAUCCGAAGGAGGACAGAACGUCCACUCGGAUGGCUAA
GGGAGAGCCAGUUGGGGGCCCAAGGUGAGAUGAAGCUGUAGCCUCACUGGAAGGACUAGAGGUUAGAGGAGACCCCCCCAAAACAAUGUCUAGGAAACCAGGAGGGCCCG
GAGAUAACCGGGCUGUCAAUAUGCUAGCCGUCGCGUCUUAACGCGCAGCCCUGAAGGCCCCACACCUUGUGGAUCUUGCCGUGGGUAUGUUUCUGGCAUGUGUUUCUCAA
GCCUGCAACCGAAGCCGAACAGCCACAUGAACAGUUUGAGCGUGGUAGCGCUGUGUGAGUUGGCGGUGGAUCCCCCUCGUGGUAACACGAGCCCCCGUGGCCAAAAGCCC
AGUGUUUACAGCACCUCUCACAUCCAGGACGACCCCAUCCUGGCGCUCACUCUUAGUAGUAUGGCUUAGUACGCAUUAGGUGGUAAGCCGAGCUCUCCCUCGGCCUUGUU
CUGAAUGCACACAUGUCUAGGGGCUAAGGAUGUCCUACAGGUACCCGCACGUAACCUUCAGAGAGUGCGGAUCUGAGUAGGAGACCGUGGUGCACUGCUUUACAGAUGCA
GCCCGGUUUAAAAAGCGUCUAUGCCCCUACAGGGUAGCGGUGGGCCGCGCCCUUUCCUUUUAAAACUACUUGUUCCUCCGUGUAGCUCAGUUUGGUAGAGCGCCUGAUUU
GGGAUCAGGAGGUCCAAGGUUCAAAUCCUUGUAUGGAGACAAAAGUUUGGGCUAAAGCCCACUGAUGAGCCGCUGAAAUGCGGCGAAACUUUUGGAGAGUCAGGCCGGAU
UACAUGUCCGCCACCGGAUGUUGGAUGACGGUGCUGCCUGCAGUCCAACCCCAGGAGGACUGGGUAAUUAUGGUGGGGCGUGAUGGGGAGGCUUCGGCCUGCUGGUUUCA
CUAACGCCAGUCUGUCAACCCUGUCACGUCCUGCCACCUGUAGUGUUGUGUGAUCUUGCGCGAUAAAUGCUGACGUGAAAACGUUGCGUAUUGCUACAACACUUGGUUAG
CUAUUUAGCUUUACUAAUCAAGACGCCGUCGUGCAGCCCACAAAAGUCUAGAUACGUCACAGGAGAGCAUACGCUAGGUCGCGUUGACUAUCCUUAUAUAUGACCUGCAA
AUAUAAACCAGGGAAAGACCGGAGUGGUUCUCUGCUUUUCCUCCAGAGGUCUGUGAGCACAGUUUGCUCAAGAAUAAGCAGACCUUUGGAUGACAAACACAAAACCAGGU

GCCUUUGAACCCUUUAUCCCGGGGUUCUUCGGAAGGUGAGACUUGUCCAUGUAGAGUCUCGCCGUGCACGGUAUCACACUGAUGAUACCUUCAGAGUGCAGGCAUCGCCU
ACGGUUUUCCGUAGGUUCCCCCUAGGGG
>0.836-0
GGAGGCGUGGCAGAGUGGUUUAAUGCACCGGUCUUGAAAACCGGCAGUCGCUCCGGCGACUCAUAGGUUCAAAUCCUAUCGCCUCCGAUGUCUAAAAAACCAGGAGGACC
CGGAAGAAACCGGGUCGUCAGUAUGGCUGCCCUUGGGUUUUACUCCUUGAACCCUUCGGAAGAACUCUUUGGAGUUCGUACCAGUACCUCACAUAGUGAGGUAAUAAGAC
UGGUGGGCAGCGCCUAGUCGAAAGACUAGGUGAUCUCUAAGGAGAGUUAUGGUGGGCUGGGCGGGGGCAUCGAAAGAUGCGCCGGUUUCCUUGAUCACCGGUUACGCCAA
CCCCGUUCAGUUCACCACCAGCACGCAGGAAUCUUAGACCAACCCUACCCGUGCAACUGCAAGUUUCGCCCGGUCUUUCCGGGUCUAGAGAGACAAACAGAUGUACUGAG
GCCAACUCCACGAUUGGUCUACUAGCGGGUACUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGCGGGAUUUCCCCCAUGGUGACAUGGACCCGCCGGGCC
AAAAGCCACGCCUCACGGCCUCAUGUGUGUGCAACCCCAGCACGGCAAUUUGUCUGUGAAACACACCUUAAGGUAACACUGAGACUGGUACUUGGUUUCUGGAGACAGGC
UAAGGAUGCCCUCCAGGUACCCCGAGGUAACACGAGACACUCGGGAUCUGAGAAGGGGACCAGGAGUUCUAUCAAACUGCCUGGUUUAAAAAGCUUCUAUGCCUGAAUAG
GUGACCGGAGGCCGGCACCUUUUCCUUUACUACACCUUAUUUCUGGGACAGACCAGAAAUCCAACUAUCUACUCAGCAUCAUUCUAGGCACAGAACGCCAGAAAAUGGAA
UGGUGCUGUUGAAUCAACAGGUUCUGUGGCCGGCAUGGCCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACGAUCCGAGGGGACUGUCCCUCUCGAGAAUCGGCAAAUGG
GGCCCCACGACGAACUGUGGCAGCACACCAUUUGGUGACGGGAAGCUGGUCGCUCCCGACGCAGGCCGUCGUAAGAACUUUGUGAGACCAAAGUCCAGUCGAGACCUGAA
GUGGGUUUACUGAUGAGGCUGUGGAGAGAGCGAAAGCUUUACUCCCACACAAGCCGAAACUGGAGUUAAGAUGUGAUCUUGCUUCCUUAUACAAUUUUGAGAGGUUAAUA
AGAAGGAAGUAGUGCUAUCUUAAUAAUUAGGUUAACUAUUUAGUUUUACUGUUCAGGAUGCCUAUUGGCAGCCCCAUAAUAUCCAGGACACCCUCUCUGCUUCUUAUAUG
AUUAGGUUGUCAUUUAGAAUAAGAAAAUAACCUGCUAACUUUCAA
>0.836-1
GACCUCGUGGCGCAACGGUAGCGCGUCUGACUCCAGAUCAGAAGGCUGCGUGUUCGAAUCACGUCGGGGUCAAUGAACCAACGAAAAAAGGUGGUUAGGCCACCUUUCAA
UAUGGGUGCCUUUGAAACCAUCUCCUAGGUUUCUUCGGAAGGACUUCGGUCCGUGUACUUCUAGCACAAGAUGCUAGUUUCAGGGUACGGGCACCCCCCCCUCAUUUAAU
GGGGGGCCUCCUCAAGGAGAAUUAUGGCAGUCCAGGCGGAGCAGCUUUCGAGCUGGCCGGUAUCCGUUGGGGCCGGUAUUCCUACCUCCGUUUGGGCUGCCACCCCACUA
GCCGUUGCGCCUUAGCGCACAGUCCUGAAGGCCCCACAUCUUGUGGAUCUUGCCGUGGACAUGUUUCUGGCAUGUGUUUCUCAAGCCUGCAACCGACGACUGAACAGACG
UGUGAACAGUUUGAGUGUGGUAGCGCUGUGUGAGUUGGCGGUGGAACCCCCUCGUGGUGACACGAGCCCCCGUGGCCAAAAGCCCAGUGUUUACAGCACCUCUCACAUCC
AGGGAAACCCCACCCUGGCGCUCACUCUUAGUAACACUUCCUAGUAGGUAUUAGGUGGUAAGCCGAGCUCUUCCUCGGUCUUGUUCUGAAUGCACACAUGUCUAGGGGCU
AAGGAUGUCCUACAGGUACCCGCACGUAACCUUCAGAGAGUGUGGAUCUGAGUAGGAGACUGUGGUGCGCUGCUUUACAGACGUAGCCCAGUUUAAAAAACGUCUAUGCC
CCUACAGGGUCGCGGUGGGCCGCACCCUUUCCUUUUACAAACUUGGUUUUCAGGGAAAGACCGGAGUGGUUCUCUGCUUUUCCUCCAGGGGUCUGUGAGCACAGUUUGCU
CAAGAAUAAGCAGACCUUUGGAUGAAAAACACAAAACCAUCGGGUCGGCAUGGCAUCUCCACCUCCUCACGGUCCGACCUGGGCAUCCGAAGGAGGACAACGUCCCCUCG
GAUGGCUAAGGGAGAGCCAGUGAGACGGGUAUGGCAGCAGAAAACGGGGAGCUAGCCGUUCCCGACGUACUCGUCUCAUAAAAAUUUGUGAGGCUUUCGGGUGGUGUGUG
CCACCCCUGAUGAGUCCGAAAGGACGAACUGACUAUGUGAUCUUAUUAAAAUUAGGUUAAAUUUCGAGGUUAAAAAUAGUUUUAAUAUUGCUAUAGUCUUAGAGGUCUUG
UAUAUUUAUACUUACCACACAAGAUGGACCGGAGCAGCCCUCCAAUAUCUAGUGUACCCUCGUGCUCGCUCAAACAUUAAGUGGUGUUGUGCGAAAAGAAUCUCACUUCA
AGAAAAAGAA
>0.897-0
AUGUCUAAAAAACCAGGAGGACCCGGAAGAAACCGGGUCGUCAGUAUGUGGCAGAUGGCGUGCCAUAACUCUACUAGUGGGAUACCACGCUUGUGGACCUUAUGCCCACA
CAGCCAUCCUCUAGUAAGUUUGUAAGACGUCUGAUGACGUGUGGGAAAACACUAGAAAUAUUAACUUGCUGUAAAGCAUCCUACUGCCAGCGGAAUCACAUCUGGUAACA
GAUGCCUCUGGGGCCAAAAGCCAAGGUUUAGCAGACCCUUUAGGAUUGGUUCAAUACCUGAGUUAAUAUGGAAAGUGCUUAGUACCUGCUGACUUGGUAGUAGUGCAAAC
ACUAGUUGUAUGGCCCGCGAAGGAUGCCCAGAAGGUACCCGUAGGUAACAAGUGACACUAUGGAUCUGAUCUGGGGCUGGAUACCUCUAUCUUGGUGAUCCAGUUAAAAA
ACGUCUAGCGGGCCAAGCCCAGGGGGGAUCCCUGGUUUCCCUAUUUUUAUUAUUAUCACGUCCCGCUGGUGUAAUGGAUAGCAUACGAUCCUUCUAAGUUUGCGGUCCUG
GUUCGAUCCCAGGGCGGGAUACAAAAGUCUGGGCUAAGCCCACUGAUGAGUUGCUGAGAUGCGACGAAACUUUUGCAGGGAAAGACCGGAGUGGUUCUCUGCUUUUCCUC
CAGGGGUCUGUGAGCACAGUUUGCUCUAGAAGAAGCAGACCUUUGGAUGAAAAACACAAAACCAAGGGGAUGGUGUGGCAGCGCACAACGACAUCGUGACGGGAGUGGGU
CGCCCCCGACGCACCAUCCUCUUGGGAAAAAUUUUCGUGAGACCGCUGCCCUUGGGUUUUACUCCUUGAACCCUUCGGAAGAACUCUUUGGAGUUCGUACCAGUACCUCA
CAUAGUGAGGUAAUAAGACUGGUGGGCAGCGCCUAGUCGAAAGACUAGGUGAUCUCUAAGGAGGUGGCCGGCAUGGCCCCAGCCUCCUCGCUGGCGCCGGCUGGGCAACG
AUCCGAGGGGACUGUCCCUCUCGAGAAUCGGCAAAUGGGGCCCC
>0.897-1
AUGAACAACCAACGGAAAAAGACGGGUAACCCGUCUUUCAAUAUGAAGCAGGUUUCUCCAACUGACACAAACCGUGCAACUUGAAACUCCGCCUGGUCUUUCCAGGUCUA
GAGGGGUGACACUUUGUACUGUGUUUGGCUCCACGUUCGAUCCACUGGCGAGUGUUAGUAACACCACUGCUGCUUCGUAGCGGAGCAUGACGGCCGUGGGAACUCCUCCU
UGGUAACAAGGACCCACGGGGCCGAAAGCCACGUCCGAAUGGACCCGCCAUGUGUGCAACCCCAGCACGGUAGCUUUGUUGUGAAACCCACUUUAAAGUGACAUUGAUAC
UGGUACCCAAAUACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGCGACACUCGGGAUCUGAGAAGGGGACUGGGGCUUCUGUAAAAGCGCCCAGU
UUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCUCUAACUACUUUGACUCUGACCUCGUGGCGCAACGGUAGCGCGUCUGACUCCAGAUCAGAA
GGCUGCGUGUUCGAAUCACGUCGGGGUCAACCGGAUGUGCUUUCCGGUCUGAUGAGUCCGUGAGGACGAAACAGGACCCAGGAAACUGGGGGGGCGGUUCUUGUUCUCCC
UGAGCUACCACCAUCCAGGCACAGAUAGCCUGACAAGGAGAUGGUGUGUUGCGAGUGAGGGCAACUCUGGGAUUAGCUCAAUGGGUGUGACGACCCUACCCUUCCGCAUU
UGUAAAUAAUUGAGCCAGUCAGUUGCCUUUGAGAGUUACUCCUUGCUCUCUUCAGAAGUUCUCUACCGAGAACGUGCAGUGGCUGUCAUAGACAGCCUUCAGACUGCGGG
CAACGUCAGUUCCAUAAGGAACUGAUAAUCUCCAAAAGGAGUCGGGUCGGCAUGGCAUCUCCACCUCCCCGUGGUCCGACCUGGGCAUCCGAAGGAGGACAAACGUCCAC
UCGGAUGGCUAAGGGAGAGCCAGU
>0.944-0
UGGGAAUUAGCCAAGUUGGUAAGGCAUAGCACUUUGACUGCUAGAUGCAAAGGUUCGAGUCCUUUAUUCCCAGGUCAAUGGUAGCUCAGGCGGGGCAGCCUCCGGGCUGG
CCGGUAUCCUGUGAAGCCGGUAUUCUCACCCCCGUCUGGGCUAUCGCCAUCAUGUCUAAAAAACCAGGAGGACCCGGGAAGCCCCGGGUCGUCAAUAUGACCUGUAUUAU
GUACAGGGUGGCAGAUGGCGUGCCAUAAAUCUAUUAGUGGGAUACCACGCUUGUGGACCUUAUGCCCACACAGCCAUCCUCUAGUAAGUUUGUAAAAUGUCUGGUGAGAU
GUGGGAACUUAUUGGAAACAACAAUUUGCUUAAUAGCAUCCUAGUGCCAGCGGAACAACAUCUGGUAACAGAUGCCUCUGGGGCCAAAAGCCAAGGUUUGACAGACCCAU
UAGGAUUGGUUUCAAAACCUGAAUUGUUGUGGAAGAUAUUCAGUACCUAUCAAUCUGGUAGUGGUGCAAACACUAGUUGUAAGGCCCACGAAGGAUGCCCAGAAGGUACC
CGCAGGUAACAAGAGACACUGUGGAUCUGAUCUGGGGCCAACUACCUCUAUCAGGUGAGUUAGUUAAAAAACGUCUAGUGGGCCAAACCCAGGGGGGAUCCCUGGUUUCC
UUUUAUUGUUAAUAUUGACAAAACCUCCCAGAGAAGGCCGAACUGGGAGGCCAUGAAGGAAGAGUCUGUGCUAAGCACACUGACGAGUCUCUGAAAUGAGACGAAACUCU
UUCAGGGAAAGACCGGAGUGGUUCUCUGCUUUUCCUCCAGAGGUCUGUGAGCACAGUUUGCUCAAGAAUAAGCAGACCUUUGGAUGACAAACACAAAACCAAGGGCGCAU
UCCGGCAGCACACCAGUGAGAGUGGUGACGGGAAACUGGUCACUCCCGACGGAGCUGCGCCUUGUGAAACUUUGUGAGACCCCUGUGGCCGGCAUGGCCCCAGCCUCCUC

GCUGGCGCCGGCUGGGCAACGAUCCGAGGGGACUAUCCCUCUCGAGAAUCGGCAAAUGGGGCCCC
>0.944-1
CUCCGUGUAGCUCAGUUUGGUAGAGCGCCUGAUUUGGGAUCAGGAGGUCCAAGGUUCAAAUCCUUGUAUGGAGAGUCAAUGAUGAACUGGUUGAGGGAUCGAAAGAUCGC
CGUUUACACCGUGUUACGGUACGCCAACCUUGAUCAGUUCAUCACCAGUAUGAACAACCAACGGAAAAAGACGGCUCGACCGUCUUUCAAUAUGACGCAGGUUUCCUCAA
CUGACACAACUCGUGCAAACUGUAACCCCGCCUGGUCUUUCCAGGUCUGGAGGGGAAGCUUUUGUACUGUGUUUGGCUCCACGUUCGGUCUACUGGCGAGCGCUAGUAGU
GGCAAUGUUGUUUCGUAGCGGAGCAUGUUGGCCGUGGGAAUCCCUCCUUGGCGACAAGGACCCACGGGGCCGCAAGCCACGUCUUUGACCCAACAUGUGUGCAACCCCAG
CACGGCAACCUUGCCACGAAAACCACUUCAAGGUAACACUGAUACUGGUACUCAAAUACUGGUGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGGGACAC
UUGGGAUCUGAGAAGGGGAUUGGGACUUCUAUAAAAGUGCCCAAUUUAAAAAGCUUCUAUGCCUGAAUAGGUGACCGGAGGCCGGCACCUUUCCUCUACAACUACUUUAC
CCAAAAGGCCUCCUGGAAGGCUCACCAGGAGUUAGGCCAUUCUAGAGGGGUGGUGUGUACCAUCCCUGAUGAGUCCAAAAGGACGAAAUGGCCAGGAAACUGGGGGGGCG
GUUCUUGUUCUCCCUGAGCCACCGCCAUCCAGGCACAGAUAGCCUGACAAGGAGAUGGUGAGUGACUCGGAAAAACACCCGCUGAGACGGGUAUGGCAGCAGAAAACGGG
GAGCUAGCCGUUCCCGACGUACUCGUCUCAUAAAAAUUUGUGAGGCUUUCUCGGGUCGGCAUGGCAUCUCCACCUCCUCGCGGUCCGACCUGGGCAUCCGUAGGAGGACA
GACGUCCUACUCGGAUGGCUAAGGGAGAGCCAGU
>0.997-0
UGUCCGUAGUGGAUGUGUAUCCACUCUGAUGAGUCCAAAAGGACGAAACGGAUGGUCAAUGGUAGCUCAGGCGGGGCAGCCUCCGGGCUGGCCGGUAUCCUGUGAAGCCG
GUAUUCUCACCCCCGUCUGGGCUAUCGCCAUCUGGCAGAUGGCGUGCCAUAACUCUACUAGUGGGAUACCACGCUUGUGGACCUUAUGCCCACACAGCCAUCCUCUAGUA
AGUUUGUAAGACGUCUGAUGACGUGUGGGAAAACACUAGAAAUAUUAACUUGCUGUAAAGCAUCCUACUGCCAGCGGAAUCACAUCUGGUAACAGAUGCCUCUGGGGCCA
AAAGCCAAGGUUUAGCAGACCCUUUAGGAUUGGUUCAAUACCUGAGUUAAUAUGGAAAGUGCUUAGUACCUGCUGACUUGGUAGUAGUGCAAACACUAGUUGUAUGGCCC
GCGAAGGAUGCCCAGAAGGUACCCGUAGGUAACAAGUGACACUAUGGAUCUGAUCUGGGGCUGGAUACCUCUAUCUUGGUGAUCCAGUUAAAAAACGUCUAGCGGGCCAA
GCCCAGGGGGGAUCCCUGGUUUCCCUAUUUUUAUUAUUAUCACCAGGGAAAGACCGGAGUGGUUCUCUGCUUUUCCUCCAGGGGUCUGUGAGCACAGUUUGCUCUAGAAG
AAGCAGACCUUUGGAUGAAAAACACAAAACCAAUGAAUGACCAACGGAAAGAGGCGAAAAACACGCCAUUCAAUAUGGCUGCCCUUGGGUUUUACUCCUUGAACCCUUCG
GAAGAACUCUUUGGAGUUCGUACCAGUACCUCACAUAGUGAGGUAAUAAGACUGGUGGGCAGCGCCUAGUCGAAAGACUAGGUGAUCUCUAAGGAGGCGGAUAUCGUAUA
AUGGUAUUACCUCAGACUUCCAAUCUGAUGAUGUGAGUUCGAUUCUCAUUAUCCGCUGCAAGGUACGGAUUAGCCGUAGGGGCUUGAGAACCCCCCCUCCCCACUCAUUU
UAUUUCC
>0.997-1
CAAAAGUCUGGGCUAAGCCCACUGAUGAGCUGCUGAAAUGCGGCAAAACUUUUGAAUUAUGGUGGGGCGUACAGGGGCAUCGCAAGAUGCGCCGGUAAGGGUAGCCGCCG
GUAACGCCAACUCUGUACGUCUCACCACCUCUACGCAGGAAUCUUAGACCAACCCGCCCGUGCAACUGCAAGUUUCGCCCGGUCUUUCCGGGUCUUGAGAGACAAACAGA
UGUACUGAGGCCAACUCCACGAUUGGUCUACUAGCGGGUACUAGUAACACUCACUUUGCUUCGUAGCGGAGCACAUGAGCGGCGGGAACACCCCCAUGGUAACAUGGACC
CGUCGGGCCAAAAGCCACGCCUCACGGCCUCAUGUGUGUGCAACCCCAGCACGGCAACUUGUUUGUGAACCACACCUUAAGGUAACACUGAGACUGGUACUUGGUUUCUG
GAGACAGGCUAAGGAUGCCCUUCAGGUACCCCGAGGUAACACGAGACACUCGGGAUCUGAGAAGGGGACCAGAAGUUCUAUCAAACUGUCUGGUUUAAAAAGCUUCUAUG
CCUGAAUAGGUGACCGGAGGCCGGCACCUUUUCCUUUACUUAAACCCAAUUCUGGGAUAGACUAGGAGAUCUUCUGCUCUGCACAACCAGCCACACGGCACAGUGCGCCG
ACAAUGGUGGCUGGUGGUGCGAGAACAAUGUCUAAAAAACCAGGAGGACCCGGAAGAAACCGGGUCGUCAGUAUGGGCGCCUUUGAAACCAUCUCCUAGGUUUCUUCGGA
AGGACUUCGGUCCGUGUACUUCUAGCACAAGGUGCUAGUUUAGAGUACGGGCAUCCCCCCGUUUUGCGGGGGUCUCUAAAAGGAGUGGGAAUUAGCCAAGUUGGUAAGGC
AUAGCACUUUGACUGCUAGAUGCAAAGGUUCGAGUCCUUUAUUCCCAGAUGGGAUGAUGCGGCAACUCGCGAAAGCUAUGGGAAACCGGUCGCACCCGAAGCACCACCAA
UGUGGAACCGUUUCGGGAGGCC

Appendix D. Viral genome outgroup identification benchmark reults

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Tomato vein clearing leaf deformation virus	-	-	-	-	-	-
0	Tomato leaf curl Mali virus	0.296908	0.433735	0.423749	0.426529	0.254082	0.395885
	Rice latent virus 1	1.259702	0.820521	0.493540	10.000000	1.737076	10.000000
	Cotton leaf curl Multan virus	-	-	-	-	-	-
1	Desmodium mottle virus	0.421611	0.287905	0.432928	0.262633	0.152544	0.316168
	Opuntia virus 1	0.564190	0.673353	1.738342	0.815264	1.718415	0.378670
	Sweet potato virus C	-	-	-	-	-	-
2	Celery mosaic virus	0.779482	0.294969	0.386097	0.356254	0.306945	0.354209
	Rose yellow mosaic virus	0.751509	10.000000	10.000000	1.122173	10.000000	1.149298
	Tomato enation leaf curl virus	-	-	-	-	-	-
3	Chenopodium leaf curl virus	0.411580	0.507078	0.472169	0.296908	0.433735	0.296908
	Sporobolus striate mosaic virus 1	10.000000	1.544169	0.833145	1.259702	0.820521	1.259702
	Tomato leaf curl Pakistan virus	-	-	-	-	-	-
4	Coccinia mosaic Tamil Nadu virus	0.433735	0.423749	0.426529	0.254082	0.395885	0.421611
	Sporobolus striate mosaic virus 1	0.820521	0.493540	10.000000	1.737076	10.000000	0.564190
	Ocimum mosaic virus	-	-	-	-	-	-
5	Oxalis yellow vein virus	0.287905	0.432928	0.262633	0.152544	0.316168	0.779482
	Paspalum striate mosaic virus	0.673353	1.738342	0.815264	1.718415	0.378670	0.751509
	Potato virus A	-	-	-	-	-	-
6	Chilli veinal mottle virus	0.294969	0.386097	0.356254	0.306945	0.354209	0.411580
	Caladenia virus A	10.000000	10.000000	1.122173	10.000000	1.149298	10.000000
	Vernonia yellow vein virus	-	-	-	-	-	-
7	Pouzolzia mosaic Guangdong virus	0.296908	0.433735	0.423749	0.426529	0.254082	0.395885
	Maize striate mosaic virus	1.259702	0.820521	0.493540	10.000000	1.737076	10.000000
	Okra leaf curl virus	-	-	-	-	-	-
8	Sida angular mosaic virus	0.421611	0.287905	0.432928	0.296908	0.433735	0.423749
	Eragrostis curvula streak virus	0.564190	0.673353	1.738342	1.259702	0.820521	0.493540
	Tomato leaf curl Kumasi virus	-	-	-	-	-	-
9	Blechum yellow vein virus	0.426529	0.254082	0.395885	0.421611	0.287905	0.432928
	Maize streak dwarfing virus	10.000000	1.737076	10.000000	0.564190	0.673353	1.738342
	Hollyhock leaf curl virus	-	-	-	-	-	-
10	Cotton leaf curl Multan virus	0.262633	0.152544	0.316168	0.779482	0.294969	0.386097
	Maize streak virus	0.815264	1.718415	0.378670	0.751509	10.000000	10.000000
	Sida yellow mosaic Yucatan virus	-	-	-	-	-	-
11	Tomato leaf curl Pune virus	0.356254	0.306945	0.296908	0.433735	0.423749	0.426529
	Beet severe curly top virus	1.122173	10.000000	1.259702	0.820521	0.493540	10.000000
	Rice latent virus 2	-	-	-	-	-	-
12	Oat dwarf virus	0.254082	0.395885	0.421611	0.287905	0.432928	0.262633
	Opuntia virus 1	1.737076	10.000000	0.564190	0.673353	1.738342	0.815264
	Tomato leaf curl New Delhi virus 2	-	-	-	-	-	-
13	Tomato yellow leaf curl Vietnam virus	0.152544	0.316168	0.779482	0.294969	0.386097	0.356254
	Maize streak virus	1.718415	0.378670	0.751509	10.000000	10.000000	1.122173

Continued on next page

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Cotton leaf curl Burewala virus	-	-	-	-	-	-
14	Sweet potato leaf curl Hubei virus	0.306945	0.354209	0.296908	0.433735	0.423749	0.426529
	Eragrostis streak virus	10.000000	1.149298	1.259702	0.820521	0.493540	10.000000
	Tobacco leaf curl Yunnan virus	-	-	-	-	-	-
15	Cotton chlorotic spot virus	0.254082	0.395885	0.421611	0.287905	0.432928	0.262633
	Grapevine red blotch-associated virus	1.737076	10.000000	0.564190	0.673353	1.738342	0.815264
	Cabbage leal curl virus isolate Jamaica	-	-	-	-	-	-
16	Chilli leaf curl Sri Lanka virus	0.152544	0.316168	0.779482	0.294969	0.375119	0.462984
	Panicum streak virus	1.718415	0.378670	0.751509	10.000000	10.000000	0.497553
	Synedrella yellow vein clearing virus	-	-	-	-	-	-
17	Tomato leaf curl Sinaloa virus	0.515810	0.447103	0.305838	0.460951	0.579394	0.301410
	Chickpea chlorotic dwarf Sudan virus	1.762706	10.000000	1.354539	1.763550	10.000000	1.467501
	Ageratum leaf curl betasatellite	-	-	-	-	-	-
18	Tomato yellow leaf curl virus-associated DNA beta	0.451604	0.245635	0.165687	0.421705	0.507097	0.271144
	Desmodium leaf distortion deltasatellite	10.000000	10.000000	10.000000	0.316049	10.000000	1.619082
	Zucchini tigre mosaic virus	-	-	-	-	-	-
19	Diuris virus Y	0.338625	0.461606	0.426968	0.406573	0.296723	0.834222
	Areca palm necrotic spindle-spot virus	10.000000	1.889025	10.000000	10.000000	10.000000	10.000000
	Saccharum streak virus	-	-	-	-	-	-
20	Sweetpotato symptomless mastrevirus 1	10.000000	0.375119	0.462984	0.375119	0.462984	0.515810
	Sweet potato leaf curl South Carolina virus	10.000000	10.000000	0.497553	10.000000	0.497553	1.762706
	Tomato dwarf leaf virus	-	-	-	-	-	-
21	Euphorbia leaf curl Guangxi virus	0.447103	0.305838	0.460951	0.579394	0.301410	0.451604
	Maize striate mosaic virus	10.000000	1.354539	1.763550	10.000000	1.467501	10.000000
	Kenaf leaf curl betasatellite	-	-	-	-	-	-
22	Tomato leaf curl Java betasatellite	0.245635	0.165687	0.421705	0.507097	0.271144	0.338625
	Tomato leaf curl virus satellite DNA	10.000000	10.000000	0.316049	10.000000	1.619082	10.000000
	Tobacco leaf rugose virus	-	-	-	-	-	-
23	Tomato leaf curl Uganda virus	0.461606	0.426968	0.406573	0.296723	0.375119	0.462984
	Digitaria ciliaris striate mosaic virus	1.889025	10.000000	10.000000	10.000000	10.000000	0.497553
	Tomato leaf curl Taiwan virus	-	-	-	-	-	-
24	Sweet potato leaf curl Sichuan virus 1	0.515810	0.447103	0.305838	0.460951	0.579394	0.301410
	Axonopus compressus streak virus	1.762706	10.000000	1.354539	1.763550	10.000000	1.467501
	Actinidia virus A	-	-	-	-	-	-
25	Grapevine virus A	0.451604	0.375119	0.462984	0.515810	0.447103	0.305838
	Hydrangea chlorotic mottle virus	10.000000	10.000000	0.497553	1.762706	10.000000	1.354539
	Malvastrum yellow mosaic Helshire virus	-	-	-	-	-	-
26	Pepper yellow leaf curl Aceh virus	0.460951	0.579394	0.301410	0.451604	0.245635	0.165687
	Spinach severe curly top virus	1.763550	10.000000	1.467501	10.000000	10.000000	10.000000
	Centrosema yellow spot virus	-	-	-	-	-	-
27	Tomato vein clearing leaf deformation virus	0.421705	0.507097	0.271144	0.338625	0.461606	0.426968
	Urochloa streak virus	0.316049	10.000000	1.619082	10.000000	1.889025	10.000000
	Maize striate mosaic virus	-	-	-	-	-	-
28	Maize streak Reunion virus	0.375119	0.462984	0.515810	0.447103	0.305838	0.460951
	Tomato mottle Taino virus	10.000000	0.497553	1.762706	10.000000	1.354539	1.763550
	Sida yellow mosaic Alagoas virus	-	-	-	-	-	-
29	Tomato leaf curl Gujarat virus	0.579394	0.301410	0.451604	0.245635	0.165687	0.421705
	Polygala garcinii associated virus	10.000000	1.467501	10.000000	10.000000	10.000000	0.316049
	Cherry twisted leaf associated virus	-	-	-	-	-	-
30	Cherry green ring mottle virus	0.507097	0.271144	0.338625	0.461606	0.426968	0.406573
	Chrysanthemum virus B	10.000000	1.619082	10.000000	1.889025	10.000000	10.000000
	Sweet potato leaf curl China Henan virus	-	-	-	-	-	-
31	Ramie yellow mosaic virus	0.375119	0.462984	0.515810	0.447103	0.305838	0.460951
	Euphorbia caput-medusae latent virus	10.000000	0.497553	1.762706	10.000000	1.354539	1.763550

Continued on next page

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Chickpea chlorosis Australia virus	-	-	-	-	-	-
32	Rice latent virus 2	0.579394	0.301410	0.451604	0.245635	0.165687	0.421705
	Alfalfa leaf curl virus	10.000000	1.467501	10.000000	10.000000	10.000000	0.316049
	Tomato leaf curl Ranchi virus	-	-	-	-	-	-
33	Bean bushy stunt virus	0.507097	0.271144	0.155549	0.141140	0.178042	0.129203
	Turnip curly top virus	10.000000	1.619082	10.000000	0.226000	0.052093	10.000000
	Tomato mosaic leaf curl virus	-	-	-	-	-	-
34	Tomato leaf curl Cotabato virus	0.232384	0.136765	0.124055	0.241357	0.130604	0.201336
	Opuntia virus 1	10.000000	10.000000	0.137689	10.000000	0.149645	10.000000
	Malvastrum yellow vein Chitwan virus	-	-	-	-	-	-
35	Tomato golden mosaic virus	0.174270	0.221390	0.054607	0.184686	0.155374	0.173190
	Chickpea yellow dwarf virus	10.000000	0.097102	10.000000	10.000000	10.000000	10.000000
	Sugarcane streak Reunion virus	-	-	-	-	-	-
36	Chickpea redleaf virus	0.144228	0.089556	0.051515	0.043005	0.000000	0.155549
	Sida golden mosaic virus	10.000000	0.121249	0.120936	10.000000	10.000000	10.000000
	Malvastrum bright yellow mosaic virus	-	-	-	-	-	-
37	Tomato leaf curl Guangdong virus	0.141140	0.155549	0.141140	0.178042	0.129203	0.232384
	Sporobolus striate mosaic virus 2	0.226000	10.000000	0.226000	0.052093	10.000000	10.000000
	Tomato leaf curl Patna virus	-	-	-	-	-	-
38	Erectites yellow mosaic virus	0.136765	0.124055	0.241357	0.130604	0.201336	0.174270
	Chickpea yellows virus	10.000000	0.137689	10.000000	0.149645	10.000000	10.000000
	Tomato leaf curl Hainan virus	-	-	-	-	-	-
39	Sida angular mosaic virus	0.221390	0.054607	0.184686	0.155374	0.173190	0.144228
	Sporobolus striate mosaic virus 2	0.097102	10.000000	10.000000	10.000000	10.000000	10.000000
	Bean yellow dwarf virus	-	-	-	-	-	-
40	Sugarcane streak virus	0.089556	0.051515	0.155549	0.141140	0.178042	0.129203
	Rhynchosia yellow mosaic virus	0.121249	0.120936	10.000000	0.226000	0.052093	10.000000
	Malvastrum leaf curl Philippines virus	-	-	-	-	-	-
41	Tomato dwarf leaf virus	0.232384	0.136765	0.124055	0.241357	0.130604	0.155549
	Turnip leaf roll virus	10.000000	10.000000	0.137689	10.000000	0.149645	10.000000
	Pepper yellow leaf curl Aceh virus	-	-	-	-	-	-
42	Sweet potato leaf curl Henanzhengzhou virus	0.141140	0.178042	0.129203	0.232384	0.136765	0.124055
	Sugarcane streak virus	0.226000	0.052093	10.000000	10.000000	10.000000	0.137689
	Hibbertia virus Y	-	-	-	-	-	-
43	Chilli ringspot virus	0.241357	0.130604	0.201336	0.174270	0.221390	0.054607
	Sugarcane streak mosaic virus	10.000000	0.149645	10.000000	10.000000	0.097102	10.000000
	Rhynchosia golden mosaic Yucatan virus	-	-	-	-	-	-
44	Verbena mottle virus	0.184686	0.155374	0.173190	0.144228	0.155549	0.141140
	Camellia chlorotic dwarf-associated virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.226000
	Tomato leaf curl Bangalore virus	-	-	-	-	-	-
45	Tomato rugose yellow leaf curl virus	0.178042	0.129203	0.232384	0.136765	0.124055	0.241357
	Beet curly top virus	0.052093	10.000000	10.000000	10.000000	0.137689	10.000000
	Tomato leaf curl Ghana virus	-	-	-	-	-	-
46	Tomato leaf curl Karnataka virus 3	0.130604	0.201336	0.174270	0.221390	0.054607	0.184686
	Beet severe curly top virus	0.149645	10.000000	10.000000	0.097102	10.000000	10.000000
	Squash leaf curl Yunnan virus	-	-	-	-	-	-
47	Tomato chino La Paz virus	0.155374	0.173190	0.144228	0.089556	0.155549	0.141140
	Spinach curly top Arizona virus	10.000000	10.000000	10.000000	0.121249	10.000000	0.226000
	Blainvillea yellow spot virus	-	-	-	-	-	-
48	Melon chlorotic leaf curl virus	0.178042	0.129203	0.232384	0.136765	0.124055	0.241357
	Mulberry mosaic dwarf associated virus	0.052093	10.000000	10.000000	10.000000	0.137689	10.000000
	Tomato chlorotic leaf distortion virus	-	-	-	-	-	-
49	Tomato yellow mosaic virus	0.130604	0.201336	0.174270	0.221390	0.054607	0.184686
	Turnip leaf roll virus	0.149645	10.000000	10.000000	0.097102	10.000000	10.000000

Continued on next page

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Euphorbia mosaic Peru virus	-	-	-	-	-	-
50	Corchorus yellow spot virus	0.292184	0.349613	0.315534	0.302080	0.281225	0.338885
	Grapevine red blotch-associated virus	0.501304	0.353511	0.348245	0.487338	0.502644	0.498631
	Dolichos yellow mosaic virus	-	-	-	-	-	-
51	Tomato yellow leaf curl China virus	0.318007	0.246450	0.312410	0.194391	0.150629	0.297080
	Bromus catharticus striate mosaic virus	0.360459	0.542366	0.474710	0.498810	0.581748	0.304945
	Tomato yellow leaf curl Shuangbai virus	-	-	-	-	-	-
52	Erectites yellow mosaic virus	0.431134	0.240269	0.264658	0.345359	0.305747	0.310267
	Chickpea yellow dwarf virus	0.496319	0.500637	0.514002	0.455591	0.505903	0.447658
	Tomato leaf curl New Delhi virus	-	-	-	-	-	-
53	Tomato leaf curl Arusha virus	0.202476	0.349351	0.424065	0.292184	0.349613	0.292184
	Eleusine indica associated virus	0.509375	0.341263	0.331787	0.501304	0.353511	0.501304
	Ageratum yellow vein China virus	-	-	-	-	-	-
54	Allamanda leaf curl virus	0.349613	0.315534	0.302080	0.281225	0.338885	0.318007
	Paspalum dilatatum striate mosaic virus	0.353511	0.348245	0.487338	0.502644	0.498631	0.360459
	Ageratum yellow vein Hualian virus	-	-	-	-	-	-
55	Dolichos yellow mosaic virus	0.246450	0.312410	0.194391	0.150629	0.297080	0.431134
	Axonopus compressus streak virus	0.542366	0.474710	0.498810	0.581748	0.304945	0.496319
	Ipomoea yellow vein virus	-	-	-	-	-	-
56	Pavonia yellow mosaic virus	0.240269	0.264658	0.345359	0.305747	0.310267	0.202476
	Eragrostis curvula streak virus	0.500637	0.514002	0.455591	0.505903	0.447658	0.509375
	Sweet potato leaf curl Spain virus	-	-	-	-	-	-
57	Mesta yellow vein mosaic virus	0.292184	0.349613	0.315534	0.302080	0.281225	0.338885
	Alfalfa leaf curl virus	0.501304	0.353511	0.348245	0.487338	0.502644	0.498631
	Clover yellow vein virus	-	-	-	-	-	-
58	Apium virus Y	0.318007	0.246450	0.312410	0.292184	0.349613	0.315534
	Ryegrass mosaic virus	0.360459	0.542366	0.474710	0.501304	0.353511	0.348245
	Tetterwort vein chlorosis virus	-	-	-	-	-	-
59	Diodia vein chlorosis virus	0.302080	0.281225	0.338885	0.318007	0.246450	0.312410
	Rose leaf rosette-associated virus	0.487338	0.502644	0.498631	0.360459	0.542366	0.474710
	Potato virus A	-	-	-	-	-	-
60	Onion yellow dwarf virus	0.194391	0.150629	0.297080	0.431134	0.240269	0.264658
	Yellow oat-grass mosaic virus	0.498810	0.581748	0.304945	0.496319	0.500637	0.514002
	Euphorbia mosaic Venezuela virus	-	-	-	-	-	-
61	Ageratum yellow vein Hualian virus	0.345359	0.305747	0.292184	0.349613	0.315534	0.302080
	Eragrostis curvula streak virus	0.455591	0.505903	0.501304	0.353511	0.348245	0.487338
	Rhynchosia golden mosaic virus	-	-	-	-	-	-
62	Tomato yellow leaf curl Kanchanaburi virus	0.281225	0.338885	0.318007	0.246450	0.312410	0.194391
	Chickpea chlorotic dwarf Sudan virus	0.502644	0.498631	0.360459	0.542366	0.474710	0.498810
	Beet mosaic virus	-	-	-	-	-	-
63	Catharanthus mosaic virus	0.150629	0.297080	0.431134	0.240269	0.264658	0.345359
	Narcissus latent virus	0.581748	0.304945	0.496319	0.500637	0.514002	0.455591
	Pouzolzia mosaic Guangdong virus	-	-	-	-	-	-
64	Cabbage leal curl virus isolate Jamaica	0.305747	0.310267	0.292184	0.349613	0.315534	0.302080
	Sorghum arundinaceum associated virus	0.505903	0.447658	0.501304	0.353511	0.348245	0.487338
	Tobacco leaf curl Zimbabwe virus	-	-	-	-	-	-
65	Tobacco leaf curl Pusa virus	0.281225	0.338885	0.318007	0.246450	0.312410	0.194391
	Grapevine geminivirus A	0.502644	0.498631	0.360459	0.542366	0.474710	0.498810
	Cardamom mosaic virus	-	-	-	-	-	-
66	Chinese yam necrotic mosaic virus	0.150629	0.297080	0.431134	0.240269	0.125000	0.047619
	Kalanchoe mosaic virus	0.581748	0.304945	0.496319	0.500637	10.000000	0.500000
	Cotton leaf curl Kokhran virus	-	-	-	-	-	-
67	Passionfruit severe leaf distortion virus	1.000000	0.176471	0.114286	0.500000	1.000000	0.086957
	Rice latent virus 1	1.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Potato yellow mosaic Panama virus	-	-	-	-	-	-
68	Pavonia yellow mosaic virus	0.000000	0.070796	0.040486	0.363636	10.000000	0.250000
	Bromus catharticus striate mosaic virus	10.000000	10.000000	10.000000	0.147059	10.000000	10.000000
	Croton yellow vein mosaic virus	-	-	-	-	-	-
69	Eclipta yellow vein virus	0.054545	0.125000	0.117647	0.259259	0.032787	10.000000
	Eragrostis streak virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Jatropha mosaic Nigerian virus	-	-	-	-	-	-
70	Hollyhock leaf crumple virus	10.000000	0.125000	0.047619	0.125000	0.047619	1.000000
	Bromus catharticus striate mosaic virus	10.000000	10.000000	0.500000	10.000000	0.500000	1.000000
	Dalechampia chlorotic mosaic virus	-	-	-	-	-	-
71	Pepper yellow leaf curl Indonesia virus	0.176471	0.114286	0.500000	1.000000	0.086957	0.000000
	Beet curly top Iran virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Jacquemontia yellow vein virus	-	-	-	-	-	-
72	Tomato leaf curl Gandhinagar virus	0.070796	0.040486	0.363636	10.000000	0.250000	0.054545
	Eleusine indica associated virus	10.000000	10.000000	0.147059	10.000000	10.000000	10.000000
	Tobacco leaf curl Kochi virus	-	-	-	-	-	-
73	Tobacco leaf curl Zimbabwe virus	0.125000	0.117647	0.259259	0.032787	0.125000	0.047619
	French bean severe leaf curl virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.500000
	Duranta leaf curl virus	-	-	-	-	-	-
74	Tomato leaf curl Comoros virus	1.000000	0.176471	0.114286	0.500000	1.000000	0.086957
	Sporobolus striate mosaic virus 1	1.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Lycianthes yellow mosaic virus	-	-	-	-	-	-
75	Tomato mosaic severe dwarf virus	0.000000	0.125000	0.047619	1.000000	0.176471	0.114286
	Turnip leaf roll virus	10.000000	10.000000	0.500000	1.000000	10.000000	10.000000
	Passionfruit leaf distortion virus	-	-	-	-	-	-
76	Lisianthus enation leaf curl virus	0.500000	1.000000	0.086957	0.000000	0.070796	0.040486
	Eragrostis curvula streak virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Tomato vein clearing leaf deformation virus	-	-	-	-	-	-
77	Jatropha mosaic India virus	0.363636	10.000000	0.250000	0.054545	0.125000	0.117647
	Digitaria ciliaris striate mosaic virus	0.147059	10.000000	10.000000	10.000000	10.000000	10.000000
	Rhynchosia golden mosaic virus	-	-	-	-	-	-
78	Sweet potato leaf curl Sao Paulo virus	0.125000	0.047619	1.000000	0.176471	0.114286	0.500000
	Tomato pseudo-curly top virus	10.000000	0.500000	1.000000	10.000000	10.000000	10.000000
	Sugarcane streak Nile virus	-	-	-	-	-	-
79	Chickpea chlorotic dwarf virus	1.000000	0.086957	0.000000	0.070796	0.040486	0.363636
	Malvastrum yellow vein Lahore virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.147059
	Sida yellow mosaic Yucatan virus	-	-	-	-	-	-
80	Malvastrum yellow vein Lahore virus	10.000000	0.250000	0.054545	0.125000	0.117647	0.259259
	Eragrostis minor streak virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Ageratum Yellow vein China virus - OX1	-	-	-	-	-	-
81	Mesta yellow vein mosaic Bahraich virus	0.125000	0.047619	1.000000	0.176471	0.114286	0.500000
	Grapevine geminivirus A	10.000000	0.500000	1.000000	10.000000	10.000000	10.000000
	Tomato leaf curl Cebu virus	-	-	-	-	-	-
82	Asystasia begomovirus 1	1.000000	0.086957	0.000000	0.070796	0.040486	0.363636
	Sweetpotato symptomless mastrevirus 1	10.000000	10.000000	10.000000	10.000000	10.000000	0.147059
	Tomato yellow leaf curl Vietnam virus	-	-	-	-	-	-
83	Corchorus yellow vein mosaic virus	10.000000	0.250000	0.369115	0.412759	0.799876	0.294512
	Pepper yellow dwarf virus - Mexico	10.000000	10.000000	10.000000	0.522031	1.211690	10.000000
	Hemidesmus yellow mosaic virus	-	-	-	-	-	-
84	Pavonia mosaic virus	0.272113	0.405679	0.874588	0.268683	0.269310	0.192190
	Sugarcane chlorotic streak virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Macroptilium common mosaic virus	-	-	-	-	-	-
85	Okra leaf curl virus	0.153331	0.352542	0.686401	0.237566	0.188730	0.533099
	Spinach curly top Arizona virus	10.000000	0.265475	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.1 Species-Genus outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Tomato yellow leaf curl Thailand virus	-	-	-	-	-	-
86	Soybean blistering mosaic virus	0.378072	0.314541	0.171773	0.912364	1.087700	0.338753
	Plantago lanceolata latent virus	10.000000	10.000000	10.000000	10.000000	1.513090	10.000000
	Basella rugose mosaic virus	-	-	-	-	-	-
87	Tobacco mosqueado virus	0.353273	0.270525	0.375586	0.759799	0.337818	0.270756
	Artichoke latent virus	0.500351	10.000000	0.499466	10.000000	10.000000	10.000000
	Chloris striate mosaic virus	-	-	-	-	-	-
88	Chickpea chlorotic dwarf Sudan virus	0.385051	0.889040	0.283553	0.243428	0.191619	0.142908
	Sida mottle virus	10.000000	1.843390	10.000000	0.683140	10.000000	10.000000
	African cassava mosaic virus	-	-	-	-	-	-
89	Macroptilium bright mosaic virus	0.323731	10.000000	0.242724	0.226134	0.559395	0.363579
	Maize streak virus	0.310957	10.000000	10.000000	10.000000	10.000000	10.000000
	Rice necrosis mosaic virus	-	-	-	-	-	-
90	Oat mosaic virus	0.316667	0.171452	0.301863	0.337182	0.781963	0.355218
	Apium virus Y	10.000000	10.000000	10.000000	0.462175	10.000000	10.000000
	Sauropus leaf curl virus	-	-	-	-	-	-
91	Blechum interveinal chlorosis virus	0.285581	0.345242	0.789349	0.314096	0.212337	0.328296
	Digitaria didactyla striate mosaic virus	10.000000	10.000000	1.202290	10.000000	10.000000	10.000000
	Mungbean yellow mosaic virus	-	-	-	-	-	-
92	Mungbean yellow mosaic virus	0.355180	0.805229	0.326489	0.281352	0.343374	0.707245
	Eragrostis minor streak virus	0.473915	1.209540	10.000000	10.000000	10.000000	10.000000
	Tomato leaf curl Cebu virus	-	-	-	-	-	-
93	Rose leaf curl virus	0.310383	0.255110	0.207891	0.145523	0.330210	10.000000
	Digitaria didactyla striate mosaic virus	10.000000	0.839774	10.000000	10.000000	0.295561	10.000000
	Tomato latent virus	-	-	-	-	-	-
94	Potato yellow mosaic virus	0.229195	0.209198	0.587025	0.295616	0.351289	0.355753
	Chickpea chlorosis virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.443947
	Bean chlorotic mosaic virus	-	-	-	-	-	-
95	Potato yellow mosaic Panama virus	0.913814	0.311014	0.316803	0.402458	0.859472	0.323251
	Chickpea redleaf virus	1.054980	10.000000	10.000000	10.000000	10.000000	10.000000
	Sida angular mosaic virus	-	-	-	-	-	-
96	Sweet potato leaf curl Henan virus	0.282405	0.199814	0.136116	0.411085	10.000000	0.256247
	Turnip curly top virus	10.000000	10.000000	10.000000	0.251486	10.000000	10.000000
	African eggplant mosaic virus	-	-	-	-	-	-
97	Onion yellow dwarf virus	0.213509	0.443118	0.343185	0.343979	0.336125	0.367620
	Sweet potato mild mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.475754
	Eragrostis minor streak virus	-	-	-	-	-	-
98	Bean yellow dwarf virus	0.784867	0.331394	0.283377	0.322420	0.926329	0.269417
	Asystasia begomovirus 1	1.257060	10.000000	10.000000	10.000000	1.336030	10.000000
	Pepper leaf curl Yunnan virus	-	-	-	-	-	-
99	Pepper huasteco yellow vein virus	0.287556	0.202173	0.143165	0.336363	0.715922	0.260509
	Tomato pseudo-curly top virus	10.000000	10.000000	10.000000	0.300880	10.000000	10.000000

Values returned by programs that were larger than 10.0 sub/bp, not-a-number values, or error codes are presumed to indicate a maximum level of divergence, and are represented by 10.0 sub/bp. MoM: Mottle-Map, BM2: BWA-Mem2, Swp: Swipe, Msh: Mash, CoP: Co-Phylog, Sls: Slope-SPAM

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Garlic mite-borne filamentous virus	-	-	-	-	-	-
0	Euonymus yellow vein virus	0.443073	0.470460	10.000000	0.822931	0.329755	0.424935
	Arracacha virus V	10.000000	1.370112	10.000000	10.000000	1.773937	10.000000
	Narcissus common latent virus	-	-	-	-	-	-
1	Actinidia virus A	0.762870	0.382707	10.000000	0.478963	0.624367	0.553577
	Citrus yellow vein clearing virus	0.820526	10.000000	10.000000	10.000000	1.274286	10.000000
	Tamus red mosaic virus	-	-	-	-	-	-
2	Turtle grass virus X	0.390325	10.000000	0.540896	0.933228	0.385195	1.095903
	Red clover vein mosaic virus	10.000000	10.000000	10.000000	1.940407	10.000000	10.000000
	Hop mosaic virus	-	-	-	-	-	-
3	Potato virus P	0.002496	0.761407	0.692096	0.362007	0.534451	0.002498
	Pitaya virus X	10.000000	10.000000	10.000000	10.000000	1.482090	10.000000
	Phaius virus X	-	-	-	-	-	-
4	Alternanthera mosaic virus	0.558722	0.433133	0.426016	0.921333	0.436428	0.661495
	Caucasus prunus virus	10.000000	10.000000	1.556974	1.874427	10.000000	10.000000
	Plum bark necrosis and stem pitting-associated virus	-	-	-	-	-	-
5	Beet pseudoyellows virus	0.588211	0.435853	0.927319	0.412964	0.806314	0.640818
	Wasabi mottle virus	1.947339	0.817371	0.891575	1.089733	10.000000	10.000000
	Apple stem grooving virus	-	-	-	-	-	-
6	Hop latent virus	0.498096	0.478535	0.447416	0.433749	0.567313	0.002833
	Cymbidium mosaic virus	0.790908	10.000000	1.507445	10.000000	1.109591	10.000000
	Spinach latent virus	-	-	-	-	-	-
7	Cowpea chlorotic mottle virus	0.372038	0.694015	0.388604	0.550915	0.771973	0.396224
	Frangipani mosaic virus	10.000000	10.000000	1.771333	1.297967	10.000000	1.096365
	Cordyline virus 1	-	-	-	-	-	-
8	Strawberry pallidosis-associated virus	0.526352	1.583464	0.387849	0.559756	1.007568	0.608771
	Clitoria yellow mottle virus	1.031239	10.000000	10.000000	1.298757	1.514236	0.824042
	Actinidia virus X	-	-	-	-	-	-
9	Tulip virus X	0.566321	0.652883	0.451249	0.903261	0.636963	0.511714
	Potato virus P	10.000000	1.739017	10.000000	1.241112	10.000000	10.000000
	Garlic common latent virus	-	-	-	-	-	-
10	Potato latent virus	0.811141	0.440623	0.736925	0.386490	0.580211	0.476296
	Shallot virus X	1.570384	10.000000	10.000000	1.361139	1.102749	10.000000
	Apricot pseudo-chlorotic leaf spot virus	-	-	-	-	-	-
11	Elderberry carlavirus B	0.524873	0.360657	0.441790	0.680348	0.932959	1.058232
	Lily virus X	10.000000	10.000000	10.000000	1.172486	1.861939	10.000000
	Bell pepper endornavirus	-	-	-	-	-	-
12	Phaseolus vulgaris alphaendornavirus 2	10.000000	0.462168	0.402077	0.533607	0.487823	0.182728
	Raspberry leaf mottle virus	10.000000	10.000000	1.996804	10.000000	10.000000	1.860652
	Bell pepper mottle tobamovirus	-	-	-	-	-	-
13	Brugmansia mild mottle virus	0.541499	0.505315	0.560416	1.849093	0.486377	0.823723
	Tea plant line pattern virus	1.311509	1.807205	1.387625	10.000000	10.000000	10.000000
	Helleborus net necrosis virus	-	-	-	-	-	-
14	Garlic common latent virus	0.002496	0.509850	0.523378	0.455685	0.309620	0.806314
	Actinidia virus X	10.000000	3.183588	2.222226	1.794372	2.098579	1.147483
	Lilac ring mottle virus	-	-	-	-	-	-
15	Prune dwarf virus	0.273076	0.956792	0.642008	0.567313	0.489002	0.419819
	Tomato mosaic virus	2.792559	2.586633	3.637002	2.030161	1.574774	3.007162
	Grapevine virus A	-	-	-	-	-	-
16	Potato latent virus	0.430941	0.000000	0.000000	0.000000	0.696182	0.558081
	Actinidia virus X	1.821886	0.000000	0.000000	0.000000	10.000000	10.000000
	Ambrosia asymptomatic virus 1	-	-	-	-	-	-
17	Euonymus yellow vein virus	10.000000	10.000000	0.382338	10.000000	10.000000	0.496865
	Hydrangea chlorotic mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Pepper mild mottle virus	-	-	-	-	-	-
18	Crucifer tobamovirus	10.000000	0.679690	10.000000	0.772668	0.420224	10.000000
	Asparagus virus 2	10.000000	10.000000	10.000000	0.972922	10.000000	10.000000
	Grapevine leafroll-associated virus 6	-	-	-	-	-	-
19	Beet yellows virus	0.617176	1.242307	0.509890	10.000000	0.000071	10.000000
	Cape gooseberry ilarvirus 1	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Strawberry chlorotic fleck associated virus	-	-	-	-	-	-
20	Cucurbit yellow stunting disorder virus	10.000000	0.438220	0.576531	0.000365	1.278596	0.502743
	Cape gooseberry ilarvirus 1	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Ligustrum virus A	-	-	-	-	-	-
21	Asian prunus virus 2	0.507405	10.000000	0.616333	1.696945	10.000000	1.986345
	Turtle grass virus X	1.611014	10.000000	10.000000	10.000000	1.914904	10.000000
	Persimmon virus B	-	-	-	-	-	-
22	Beet pseudoyellows virus	10.000000	10.000000	0.609616	10.000000	0.660849	1.053364
	Hibiscus green spot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine virus H	-	-	-	-	-	-
23	Grapevine virus G	0.623191	0.593465	10.000000	0.000319	0.487951	10.000000
	Euonymus yellow vein virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cnidium virus X	-	-	-	-	-	-
24	Hydrangea ringspot virus	0.397327	0.622578	1.509166	0.448552	0.895229	10.000000
	Helenium virus S	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Potato virus X	-	-	-	-	-	-
25	Cymbidium mosaic virus	0.404208	1.009124	10.000000	0.890907	0.755500	0.988854
	Ribes americanum virus A	10.000000	1.652733	10.000000	10.000000	10.000000	10.000000
	Narcissus mosaic virus	-	-	-	-	-	-
26	Senna mosaic virus	0.547502	1.233735	0.896927	0.934691	10.000000	1.630787
	Gaillardia latent virus	10.000000	10.000000	1.936994	10.000000	10.000000	10.000000
	Sweet potato chlorotic stunt virus	-	-	-	-	-	-
27	Tobacco virus 1	10.000000	0.582399	10.000000	1.278689	0.640042	0.332881
	Hibiscus green spot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Blackberry virus A	-	-	-	-	-	-
28	Potato latent virus	0.594779	0.925398	10.000000	10.000000	10.000000	0.674882
	Cnidium virus X	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Boerhavia yellow spot virus	-	-	-	-	-	-
29	Cotton leaf curl Gezira virus	0.518976	10.000000	0.657500	1.790814	0.615736	0.830513
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Carnation latent virus	-	-	-	-	-	-
30	Actinidia virus B	0.767136	10.000000	0.600512	10.000000	0.000071	3.684419
	Pitaya virus X	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Beet yellow stunt virus	-	-	-	-	-	-
31	Grapevine leafroll-associated virus 2	0.549247	0.611364	0.420980	0.609616	0.304919	1.851520
	Peanut stunt virus	2.584383	3.116718	2.292949	3.893165	2.619499	10.000000
	Rice latent virus 2	-	-	-	-	-	-
32	Tomato leaf curl Toliara virus	0.830651	2.192817	0.673923	0.492091	2.296371	0.000000
	Common bean-associated gemycircularvirus	10.000000	2.530229	3.510165	10.000000	3.612406	0.000000
	Ribgrass mosaic virus	-	-	-	-	-	-
33	Tropical soda apple mosaic virus	0.000000	0.000000	0.044812	0.107409	0.052448	10.000000
	Little cherry virus 1	0.000000	0.000000	0.158586	10.000000	10.000000	10.000000
	Actinidia virus 1	-	-	-	-	-	-
34	Plum bark necrosis and stem pitting-associated virus	0.253384	0.101232	10.000000	0.149427	0.085417	0.184294
	Lilac ring mottle virus	10.000000	0.000000	10.000000	10.000000	10.000000	10.000000
	Cucumis melo endornavirus	-	-	-	-	-	-
35	Cucumis melo endornavirus	0.115776	10.000000	0.194674	10.000000	0.067765	10.000000
	Raspberry leaf mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Rehmannia virus 1	-	-	-	-	-	-
36	Diodia vein chlorosis virus	0.138156	0.039835	0.000000	10.000000	10.000000	0.212063
	Citrus leaf rugose virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Helleborus mosaic virus	-	-	-	-	-	-
37	American hop latent virus	0.083418	0.000000	0.160737	0.157489	0.192798	10.000000
	Citrus yellow vein clearing virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Soybean mild mottle virus	-	-	-	-	-	-
38	Papaya leaf curl Guangdong virus	0.172801	10.000000	10.000000	0.052242	0.115870	0.116277
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cape gooseberry ilarvirus 1	-	-	-	-	-	-
39	Spinach latent virus	0.041184	10.000000	0.165683	0.072938	0.152524	0.201364
	Barley stripe mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Hydrangea chlorotic mottle virus	-	-	-	-	-	-
40	Lily symptomless virus	0.111615	0.000000	0.227476	10.000000	0.167143	0.153717
	Euonymus yellow mottle associated virus	10.000000	0.052029	10.000000	10.000000	10.000000	10.000000
	Potato virus X	-	-	-	-	-	-
41	Cnidium virus X	0.000000	0.192223	0.046956	10.000000	0.279398	0.121110
	Grapevine virus E	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Plum bark necrosis and stem pitting-associated virus	-	-	-	-	-	-
42	Plum bark necrosis and stem pitting-associated virus	0.066248	0.084626	0.000000	0.051864	0.140401	0.054976
	Cassava Ivorian bacilliform virus	0.000000	0.088600	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 13	-	-	-	-	-	-
43	Blackcurrant-associated closterovirus 1	10.000000	0.046685	10.000000	0.149053	0.039752	0.155467
	Lychnis ringspot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine virus B	-	-	-	-	-	-
44	Prunus virus T	10.000000	10.000000	0.129645	0.241070	0.150685	0.047607
	Peach marafivirus D	10.000000	10.000000	0.000000	10.000000	10.000000	10.000000
	Wasabi mottle virus	-	-	-	-	-	-
45	Paprika mild mottle virus	10.000000	10.000000	0.048277	0.124867	0.000000	10.000000
	Strawberry chlorotic fleck associated virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Pineapple mealybug wilt-associated virus 3	-	-	-	-	-	-
46	Little cherry virus 1	0.123236	0.223012	0.000000	0.121758	0.077884	10.000000
	Maracuja mosaic virus	10.000000	10.000000	0.044793	10.000000	10.000000	10.000000
	Tobacco mosaic virus	-	-	-	-	-	-
47	Beet soil-borne virus	0.158744	10.000000	0.000000	10.000000	0.046832	0.190507
	Cape gooseberry ilarvirus 1	10.000000	10.000000	10.000000	0.094194	10.000000	0.000000
	Pineapple mealybug wilt-associated virus 1	-	-	-	-	-	-
48	Diodia vein chlorosis virus	0.212640	0.041184	0.188297	10.000000	0.000000	0.111615
	Cucumber fruit mottle mosaic virus	10.000000	0.081461	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 13	-	-	-	-	-	-
49	Bean yellow disorder virus	0.165082	0.052995	0.203532	0.000000	0.000000	0.000000
	Ribgrass mosaic virus	0.044904	0.000000	10.000000	0.000000	0.000000	0.000000
	Ribes americanum virus A	-	-	-	-	-	-
50	Prunus virus T	0.407738	0.348592	0.296111	0.391632	0.283369	0.385282
	Garlic virus E	0.387942	0.384915	0.357653	0.377808	0.396241	0.384846
	Elderberry carlavirus B	-	-	-	-	-	-
51	Carnation latent virus	0.394344	0.318572	0.308128	0.404739	0.382286	0.375823
	Cassava common mosaic virus	0.354357	0.379722	0.309731	0.520634	0.399402	0.412902
	Caucasus prunus virus	-	-	-	-	-	-
52	Grapevine virus A	0.314961	0.533502	0.339317	0.509609	0.318155	0.304434
	Euonymus yellow mottle associated virus	0.538366	0.564203	0.370581	0.552545	0.335208	0.390099
	Apple green crinkle associated virus	-	-	-	-	-	-
53	Salvia divinorum RNA virus 1	0.000000	0.304482	0.366192	0.348842	0.412730	0.000000
	Garlic virus B	0.327090	0.357631	0.414231	0.410338	0.415075	0.366993

Continued on next page

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Peanut stunt virus	-	-	-	-	-	-
54	Lilac leaf chlorosis virus	0.329544	0.326779	0.293810	0.328480	0.341254	0.536221
	Cactus mild mottle virus	0.523285	0.399437	0.386496	0.363784	0.621832	0.371689
	Blackberry yellow vein-associated virus	-	-	-	-	-	-
55	Raspberry leaf mottle virus	0.400493	0.322187	0.384990	0.381815	0.375502	0.385054
	Peanut clump virus	0.387309	0.376613	0.410680	0.379504	0.401371	0.354114
	Mirabilis jalapa mottle virus	-	-	-	-	-	-
56	Apricot vein clearing associated virus	0.365925	0.316706	0.387644	0.401909	0.378443	0.000000
	Scallion virus X	0.396800	0.343305	0.568266	0.405997	0.406076	0.370501
	Actinidia virus X	-	-	-	-	-	-
57	Pitaya virus X	0.330231	0.312298	0.295055	0.322201	0.426997	0.310617
	Potato virus S	0.407448	0.378772	0.498866	0.356011	0.434599	0.517978
	Cassava Ivorian bacilliform virus	-	-	-	-	-	-
58	Spinach latent virus	0.388509	0.518860	0.283282	0.337879	0.300969	0.378573
	Carnation necrotic fleck virus	0.406627	0.571823	0.346060	0.377678	0.313906	0.378199
	Allium virus X	-	-	-	-	-	-
59	Nerine virus X	0.386892	0.396451	0.251879	0.378497	0.383945	0.415770
	Gaillardia latent virus	0.402036	0.409222	0.318442	0.396108	0.370987	0.416754
	Sorghum arundinaceum associated virus	-	-	-	-	-	-
60	Tomato chlorotic mottle Guyane virus	0.401638	0.435564	0.340291	0.379086	0.351548	0.369756
	Common bean-associated gemycircularvirus	0.433105	0.454524	0.416810	0.411339	0.457527	0.404569
	Cherry rusty mottle associated virus	-	-	-	-	-	-
61	Actinidia virus B	0.378528	0.254827	0.381650	0.319533	0.319134	0.283937
	Indian citrus ringspot virus	0.437306	0.527107	0.432255	0.397136	0.355608	0.369493
	Citrus yellow vein clearing virus	-	-	-	-	-	-
62	White clover mosaic virus	0.302316	0.314035	0.299273	0.379699	0.375239	0.412767
	Ligustrum necrotic ringspot virus	0.366518	0.376803	0.570998	0.361864	0.404853	0.609864
	Citrus yellow vein clearing virus	-	-	-	-	-	-
63	Phaius virus X	0.359613	0.359102	0.409434	0.376739	0.329239	0.386282
	Grapevine virus H	0.409456	0.356037	0.411281	0.359635	0.387397	0.324116
	Vanilla virus X	-	-	-	-	-	-
64	Plantago asiatica mosaic virus	0.000000	0.381263	0.403968	0.352541	0.300977	0.375502
	Phlox virus M	0.351452	0.354383	0.499213	0.404202	0.455609	0.382296
	Potato latent virus	-	-	-	-	-	-
65	Apricot latent virus	0.243521	0.363425	0.394694	0.378443	0.312265	0.334370
	Garlic virus B	0.492404	0.366991	0.424381	0.405014	0.365541	0.375834
	Schlumbergera virus X	-	-	-	-	-	-
66	Lagenaria mild mosaic virus	0.421001	0.000000	0.000000	0.000000	10.000000	1.000000
	Hydrangea chlorotic mottle virus	0.450159	0.000000	0.000000	0.000000	10.000000	10.000000
	Lagenaria mild mosaic virus	-	-	-	-	-	-
67	Garlic mite-borne filamentous virus	10.000000	10.000000	0.115385	10.000000	10.000000	0.545455
	Grapevine rupestris vein feathering virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Potato virus H	-	-	-	-	-	-
68	Lettuce chordovirus 1	10.000000	10.000000	10.000000	10.000000	0.500000	10.000000
	Indian citrus ringspot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Beet virus Q	-	-	-	-	-	-
69	Beet virus Q	1.000000	10.000000	1.000000	1.000000	10.000000	10.000000
	American plum line pattern virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Raspberry leaf mottle virus	-	-	-	-	-	-
70	Pineapple mealybug wilt-associated virus 3	10.000000	0.363636	10.000000	10.000000	0.000000	0.333333
	Cucumis melo endornavirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Sweet potato leaf curl Sao Paulo virus	-	-	-	-	-	-
71	Sporobolus striate mosaic virus 1	1.000000	10.000000	1.000000	10.000000	10.000000	10.000000
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Cucumber green mottle mosaic virus	-	-	-	-	-	-
72	Nopal tobamovirus 2	10.000000	10.000000	10.000000	10.000000	1.000000	10.000000
	Cucumber mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Strawberry necrotic shock virus	-	-	-	-	-	-
73	Olive latent virus 2	10.000000	10.000000	10.000000	10.000000	0.333333	10.000000
	Drakaea virus A	10.000000	10.000000	10.000000	1.000000	10.000000	10.000000
	Nopal tobamovirus 2	-	-	-	-	-	-
74	Frangipani mosaic virus	0.280000	10.000000	0.000000	0.071429	10.000000	10.000000
	Tomato infectious chlorosis virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Potato mop-top virus	-	-	-	-	-	-
75	Indian peanut clump virus	1.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Citrus tristeza virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Okra mottle virus	-	-	-	-	-	-
76	Tobacco leaf curl Zimbabwe virus	0.600000	10.000000	10.000000	10.000000	10.000000	10.000000
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Nerine latent virus	-	-	-	-	-	-
77	Coleus vein necrosis virus	0.500000	1.000000	10.000000	10.000000	0.000000	0.046512
	Schlumbergera virus X	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Hibiscus latent Singapore virus	-	-	-	-	-	-
78	Youcai mosaic virus	0.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Fragaria chiloensis latent virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Rhynchosia yellow mosaic India virus	-	-	-	-	-	-
79	Desmodium mottle virus	0.047619	10.000000	10.000000	0.166667	0.000000	1.000000
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Potexvirus sp.	-	-	-	-	-	-
80	Shallot virus X	10.000000	10.000000	0.333333	10.000000	10.000000	10.000000
	Sweet potato C6 virus	10.000000	10.000000	10.000000	10.000000	10.000000	1.000000
	Phlox virus M	-	-	-	-	-	-
81	Diuris virus B	0.000000	1.000000	0.025641	10.000000	0.132075	10.000000
	Donkey orchid symptomless virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 4	-	-	-	-	-	-
82	Grapevine leafroll-associated virus 4	0.000000	10.000000	1.000000	0.100000	10.000000	0.000000
	Hibiscus latent Singapore virus	10.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Tomato associated geminivirus 1	-	-	-	-	-	-
83	Ageratum yellow vein virus	0.000000	0.000000	0.569615	0.740795	1.620640	10.000000
	Common bean-associated gemycircularvirus	0.000000	0.000000	10.000000	10.000000	10.000000	10.000000
	Tomato aspermy virus	-	-	-	-	-	-
84	Asparagus virus 2	0.463024	1.115300	10.000000	0.546999	1.797120	0.846586
	Little cherry virus 1	1.332020	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine virus B	-	-	-	-	-	-
85	Hop mosaic virus	10.000000	0.822085	0.289229	10.000000	0.820784	10.000000
	Garlic virus B	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Pitaya virus X	-	-	-	-	-	-
86	Euonymus yellow vein virus	0.605779	1.595580	0.017531	1.987790	1.975320	0.330594
	Jasmine virus C	10.000000	10.000000	10.000000	1.607980	10.000000	3.143900
	Fragaria chiloensis latent virus	-	-	-	-	-	-
87	Tomato aspermy virus	0.708213	0.011628	0.708094	0.615096	0.847451	10.000000
	Actinidia virus 1	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Citrus yellow vein clearing virus	-	-	-	-	-	-
88	Garlic virus D	0.679746	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine virus G	10.000000	10.000000	10.000000	1.688250	10.000000	10.000000
	Tomato golden vein virus	-	-	-	-	-	-
89	Sweet potato golden vein Korea virus	10.000000	1.183540	0.636270	1.007190	0.537200	0.585956
	Common bean-associated gemycircularvirus	10.000000	1.668650	10.000000	10.000000	10.000000	2.808500

Continued on next page

Table D.2 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Bean yellow dwarf virus	-	-	-	-	-	-
90	Tobacco leaf curl Japan virus	0.861067	0.008954	0.494088	10.000000	0.242886	0.804579
	Common bean-associated gemycircularvirus	1.000370	1.096760	10.000000	10.000000	10.000000	10.000000
	Brugmansia mild mottle virus	-	-	-	-	-	-
91	Yellow tailflower mild mottle virus	0.575639	0.409780	1.044420	10.000000	0.521759	1.158680
	Potato yellow vein virus	10.000000	10.000000	0.934473	10.000000	1.676460	10.000000
	Macroptilium mosaic Puerto Rico virus	-	-	-	-	-	-
92	Cowpea bright yellow mosaic virus	1.611640	0.946526	1.220640	10.000000	0.926524	1.340100
	Common bean-associated gemycircularvirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Nerine latent virus	-	-	-	-	-	-
93	Mirabilis jalapa mottle virus	0.984764	10.000000	10.000000	0.938970	1.327400	0.469459
	Peach marafivirus D	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Arachis pintoi virus	-	-	-	-	-	-
94	Nerine virus X	1.182300	10.000000	0.607173	0.260442	1.039120	1.487740
	Grapevine virus F	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Lettuce infectious yellows virus	-	-	-	-	-	-
95	Tomato chlorosis virus	10.000000	2.161610	10.000000	0.842959	0.538911	10.000000
	Youcai mosaic virus	10.000000	10.000000	1.370070	10.000000	10.000000	10.000000
	Panax ginseng flexivirus 1	-	-	-	-	-	-
96	Apricot pseudo-chlorotic leaf spot virus	0.708760	0.316838	0.786894	1.170080	1.033590	10.000000
	Hosta virus X	10.000000	10.000000	1.576170	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 7	-	-	-	-	-	-
97	Rehmannia virus 1	0.575330	10.000000	0.017910	10.000000	0.470792	0.772932
	Helianthus annuus alphaendornavirus	10.000000	10.000000	1.813860	10.000000	10.000000	10.000000
	Nerine virus X	-	-	-	-	-	-
98	Arachis pintoi virus	0.317385	10.000000	0.251014	10.000000	1.142770	10.000000
	Grapevine berry inner necrosis virus	10.000000	1.008730	10.000000	10.000000	10.000000	10.000000
	Blackberry virus A	-	-	-	-	-	-
99	Garlic common latent virus	0.927725	0.200253	0.430131	0.000000	0.000000	0.000000
	Alstroemeria virus x	1.387340	10.000000	10.000000	0.000000	0.000000	0.000000

Values returned by programs that were larger than 10.0 sub/bp, not-a-number values, or error codes are presumed to indicate a maximum level of divergence, and are represented by 10.0 sub/bp. MoM: Mottle-Map, BM2: BWA-Mem2, Swp: Swipe, Msh: Mash, CoP: Co-Phylog, Sls: Slope-SPAM

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Citrus yellow vein clearing virus	-	-	-	-	-	-
0	Yam virus X	0.502589	0.540675	3.107146	0.759431	1.807205	0.415819
	Nopal tobamovirus 2	3.107562	1.341603	2.883500	2.550851	4.470004	3.600511
	Foxtail mosaic virus	-	-	-	-	-	-
1	Grapevine virus A	2.729032	2.425227	0.519588	0.445356	0.535556	0.570084
	Tomato aspermy virus	1.228327	1.317799	4.152396	2.607449	3.345327	3.805479
	Lettuce chordovirus 1	-	-	-	-	-	-
2	Caucasus prunus virus	10.000000	3.024252	1.061843	0.508702	1.025383	0.505053
	Raspberry leaf mottle virus	2.121260	2.929265	2.736089	3.929961	3.097135	3.548262
	Narcissus mosaic virus	-	-	-	-	-	-
3	Phlox virus S	1.598298	2.495760	0.463896	1.445758	3.406115	2.361050
	Carrot yellow leaf virus	1.028737	2.366938	3.431947	4.173382	2.086678	1.852047
	Passion fruit mosaic virus	-	-	-	-	-	-
4	Drakaea virus A	0.622077	2.587789	2.697544	2.767609	0.820491	2.560250
	Narcissus common latent virus	3.761314	2.888328	2.709666	2.536844	2.858638	3.288025
	Turtle grass virus X	-	-	-	-	-	-
5	Shallot virus X	0.794063	2.809654	2.227896	1.914576	1.829997	0.518108
	Beet soil-borne mosaic virus	2.600347	3.219685	3.251901	2.589217	2.008347	2.790048
	Apple mosaic virus	-	-	-	-	-	-
6	Persimmon virus B	2.331214	0.931699	1.775970	0.761229	0.447318	1.745006
	Fig fleck-associated virus	1.942260	1.452105	2.829273	2.206938	2.668101	2.018019
	Red clover vein mosaic virus	-	-	-	-	-	-
7	Lettuce chordovirus 1	1.440830	1.548471	0.556118	2.516252	2.806772	1.228446
	Oryza rufipogon endornavirus	3.612327	2.843066	0.987212	3.802222	0.904487	3.966186
	Rupestris stem pitting-associated virus	-	-	-	-	-	-
8	Cherry twisted leaf associated virus	1.074617	0.582884	0.002496	1.431872	0.531001	0.485195
	Blackberry yellow vein-associated virus	2.096558	2.486895	3.627303	1.686662	2.516660	2.546448
	Grapevine leafroll-associated virus 1	-	-	-	-	-	-
9	Passion fruit mosaic virus	2.414036	1.442235	0.511523	2.142596	2.577576	2.427144
	Ligustrum virus A	3.117720	2.869380	3.126423	2.014855	3.672221	2.234766
	Asparagus virus 2	-	-	-	-	-	-
10	Tomato chlorosis virus	0.756376	0.595762	3.066875	0.789384	2.728269	2.680743
	Potato latent virus	2.474625	3.022905	1.707890	10.000000	3.501494	2.202746
	Grapevine leafroll-associated virus 13	-	-	-	-	-	-
11	Carrot yellow leaf virus	2.837156	0.359579	1.192265	3.473653	1.185711	0.926610
	Elderberry carlavirus B	2.356006	2.117228	2.426112	2.279426	3.312058	2.897793
	Garlic virus A	-	-	-	-	-	-
12	Watermelon betaflexivirus 1	2.294914	2.591976	0.528372	1.884247	1.226886	1.721183
	Lilac leaf chlorosis virus	1.819856	2.746365	3.162695	3.676835	2.759018	2.878658
	Grapevine virus E	-	-	-	-	-	-
13	Cherry green ring mottle virus	1.213716	4.770512	1.050164	0.873413	0.525934	1.836689
	Soil-borne wheat mosaic virus	3.135896	2.987725	1.804023	2.120957	10.000000	2.856753
	Apple stem grooving virus	-	-	-	-	-	-
14	Cherry rusty mottle associated virus	2.688357	2.288838	0.360560	1.536128	3.976408	0.510112
	Oat golden stripe virus	2.349672	3.489356	1.737913	2.154399	3.327179	10.000000
	Grapevine virus G	-	-	-	-	-	-
15	Cucumber vein-clearing virus	1.020882	2.774588	0.547198	1.998451	1.544084	1.252495
	Indian peanut clump virus	3.709621	2.424714	2.765880	4.296709	3.096567	2.123276
	Elderberry carlavirus A	-	-	-	-	-	-
16	Cherry green ring mottle virus	2.902765	1.226523	0.436129	10.000000	0.676627	0.628562
	Crucifer tobamovirus	1.841903	2.741000	3.210451	2.773901	2.348761	10.000000
	Cucurbit yellow stunting disorder virus	-	-	-	-	-	-
17	Grapevine leafroll-associated virus 10	3.602580	1.861073	10.000000	1.068646	10.000000	2.548165
	Strawberry mild yellow edge virus	2.389999	10.000000	3.620724	10.000000	3.259906	10.000000

Continued on next page

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Hippeastrum latent virus	-	-	-	-	-	-
18	Mume virus A	0.617146	0.447629	0.594270	2.487165	10.000000	10.000000
	Lilac leaf chlorosis virus	2.319208	4.303259	4.455440	2.854043	10.000000	5.811435
	Grapevine virus D	-	-	-	-	-	-
19	Helleborus mosaic virus	2.770644	0.991106	2.478246	1.802731	3.694721	1.874875
	Frangipani mosaic virus	10.000000	5.107063	2.805036	10.000000	2.687299	2.767667
	Crucifer tobamovirus	-	-	-	-	-	-
20	Potato yellow vein virus	0.456348	3.646056	4.632186	10.000000	0.744401	10.000000
	Heracleum latent virus	2.087452	3.303904	3.928297	10.000000	3.744531	7.294616
	Grapevine Pinot gris virus	-	-	-	-	-	-
21	Euonymus yellow vein virus	2.679749	10.000000	3.714629	10.000000	2.706176	10.000000
	Grapevine leafroll-associated virus 7	5.124820	10.000000	10.000000	10.000000	3.888110	10.000000
	Citrus tristeza virus	-	-	-	-	-	-
22	Spinach latent virus	10.000000	10.000000	10.000000	0.661457	2.779773	1.975351
	Arachis pintoi virus	10.000000	1.881479	3.842618	10.000000	3.360421	4.305189
	Hippeastrum latent virus	-	-	-	-	-	-
23	Heracleum latent virus	2.778622	2.197724	0.523768	2.157688	1.548372	10.000000
	Persea americana endornavirus	2.861887	2.487746	10.000000	3.965501	2.414786	4.843765
	Grapevine virus I	-	-	-	-	-	-
24	Arracacha virus V	2.203025	2.537562	2.697607	2.614500	3.574017	2.057100
	Broad bean mottle virus	10.000000	3.184481	10.000000	4.171564	2.614714	5.048835
	Beet soil-borne virus	-	-	-	-	-	-
25	Tomato chlorosis virus	0.000484	2.405639	2.501378	0.698882	2.301684	4.380520
	Poplar mosaic virus	10.000000	2.879920	2.897940	2.615096	10.000000	2.385849
	Atractylodes mottle virus	-	-	-	-	-	-
26	Potato virus S	0.840443	1.951303	10.000000	3.517868	2.571823	3.897829
	Lettuce infectious yellows virus	3.291372	2.507887	10.000000	2.573142	4.789435	2.922413
	Japanese soil-borne wheat mosaic virus	-	-	-	-	-	-
27	Gentian ovary ring-spot virus	4.789541	10.000000	4.720920	3.716025	3.810285	2.533910
	Ligustrum necrotic ringspot virus	3.676635	3.442850	10.000000	10.000000	4.428907	10.000000
	Potato virus P	-	-	-	-	-	-
28	Grapevine virus F	4.862854	2.436596	10.000000	2.530752	10.000000	10.000000
	Tomato necrotic streak virus	2.374304	10.000000	10.000000	1.985642	2.569057	3.863243
	Raspberry leaf mottle virus	-	-	-	-	-	-
29	Tobacco mosaic virus	1.785681	10.000000	2.175014	4.171871	0.449898	10.000000
	Peach chlorotic mottle virus	2.934063	2.498688	10.000000	10.000000	3.595396	10.000000
	Hop mosaic virus	-	-	-	-	-	-
30	Alfalfa virus S	2.720105	3.937332	0.612356	3.169806	10.000000	2.906401
	Carnation yellow fleck virus	10.000000	10.000000	2.600717	2.512659	2.549450	2.620636
	Lettuce infectious yellows virus	-	-	-	-	-	-
31	Indian peanut clump virus	1.686410	10.000000	3.223986	3.739765	2.166817	10.000000
	Garlic virus D	10.000000	2.634789	2.574533	4.830499	2.898467	6.352713
	Japanese soil-borne wheat mosaic virus	-	-	-	-	-	-
32	Gayfeather mild mottle virus	0.707476	10.000000	2.690353	2.776337	10.000000	2.208702
	Senna mosaic virus	2.533438	4.569023	10.000000	3.127470	4.042010	2.477444
	Helleborus mosaic virus	-	-	-	-	-	-
33	Chrysanthemum virus B	0.551981	3.900903	0.099415	0.171899	10.000000	10.000000
	Passion fruit mosaic virus	10.000000	4.142209	10.000000	10.000000	10.000000	0.051902
	Blueberry shock virus	-	-	-	-	-	-
34	Lagenaria siceraria endornavirus-Hubei	10.000000	0.146030	10.000000	10.000000	0.144872	0.154373
	Hosta virus X	10.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Lagenaria siceraria endornavirus-Hubei	-	-	-	-	-	-
35	Apple mosaic virus	0.201058	0.039620	0.000000	0.000000	10.000000	0.000000
	Mume virus A	10.000000	10.000000	10.000000	10.000000	0.000000	10.000000

Continued on next page

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Actinidia virus B	-	-	-	-	-	-
36	Jasmine virus C	10.000000	0.045265	0.000000	10.000000	0.154979	10.000000
	American plum line pattern virus	10.000000	10.000000	10.000000	0.088335	10.000000	10.000000
	Persimmon virus B	-	-	-	-	-	-
37	Odontoglossum ringspot virus	10.000000	10.000000	0.113665	10.000000	10.000000	10.000000
	Cowpea mild mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Blackberry virus A	-	-	-	-	-	-
38	Grapevine Red Globe virus	0.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Tomato infectious chlorosis virus	0.000000	10.000000	10.000000	10.000000	0.000000	10.000000
	Helenium virus S	-	-	-	-	-	-
39	Chrysanthemum virus B	0.000000	0.213621	10.000000	10.000000	0.052302	0.052553
	Beet pseudoyellows virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Alternanthera mosaic virus	-	-	-	-	-	-
40	Ligustrum virus A	0.189886	10.000000	10.000000	0.000000	10.000000	10.000000
	Cucumber mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Schlumbergera virus X	-	-	-	-	-	-
41	Potato latent virus	10.000000	10.000000	10.000000	0.043192	0.000000	0.154395
	Plumeria mosaic virus	0.000000	10.000000	0.088705	0.000000	10.000000	0.000000
	Cordyline virus 3	-	-	-	-	-	-
42	Lagenaria siceraria endornavirus-Hubei	10.000000	0.188419	10.000000	10.000000	0.143721	10.000000
	Alternanthera mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Alternanthera mosaic virus	-	-	-	-	-	-
43	Shallot virus X	10.000000	10.000000	10.000000	10.000000	0.000000	0.000000
	Phaseolus vulgaris alphaendornavirus 2	10.000000	10.000000	10.000000	0.000000	10.000000	10.000000
	Vicia faba endornavirus	-	-	-	-	-	-
44	Prune dwarf virus	10.000000	10.000000	10.000000	0.102197	0.049295	10.000000
	Shallot virus X	10.000000	10.000000	10.000000	0.000000	10.000000	10.000000
	Indian peanut clump virus	-	-	-	-	-	-
45	Beet yellow stunt virus	10.000000	0.119690	10.000000	10.000000	0.041319	10.000000
	Potato virus P	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Indian peanut clump virus	-	-	-	-	-	-
46	Rattail cactus necrosis associated virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Verbena latent virus	10.000000	0.000000	10.000000	10.000000	10.000000	0.000000
	Mint virus 1	-	-	-	-	-	-
47	Fragaria chiloensis latent virus	0.122171	0.000000	10.000000	10.000000	0.178413	0.000000
	Cassava virus X	10.000000	10.000000	10.000000	0.000000	10.000000	10.000000
	Strawberry chlorotic fleck associated virus	-	-	-	-	-	-
48	Pea early-browning virus	10.000000	10.000000	10.000000	10.000000	0.129647	10.000000
	Currant virus A	10.000000	10.000000	10.000000	10.000000	10.000000	0.000000
	Areca palm velarivirus 1	-	-	-	-	-	-
49	Hoya chlorotic spot virus	10.000000	10.000000	10.000000	0.000000	0.125393	10.000000
	Grapevine virus I	10.000000	0.000000	10.000000	10.000000	0.000000	10.000000
	Bell pepper endornavirus	-	-	-	-	-	-
50	Broad bean mottle virus	0.345114	0.387054	0.406978	0.395575	0.356037	0.322924
	Caucasus prunus virus	0.412448	0.407985	0.411868	0.414512	0.398646	0.324717
	Indian peanut clump virus	-	-	-	-	-	-
51	Barley stripe mosaic virus	0.354069	0.413896	0.377672	0.310058	0.360294	0.350339
	Nerine latent virus	0.417409	0.357527	0.447818	0.406962	0.365869	0.410448
	Hibiscus latent Fort Pierce virus	-	-	-	-	-	-
52	Lilac leaf chlorosis virus	0.492362	0.324414	0.471482	0.370882	0.448684	0.338189
	Clover yellow mosaic virus	0.711531	0.437307	0.373827	0.387565	0.460212	0.362689
	Amazon lily mild mottle virus	-	-	-	-	-	-
53	Strawberry pallidosis-associated virus	0.331931	0.384692	0.361267	0.335584	0.370635	0.373660
	Currant virus A	0.358778	0.373479	0.465582	0.369158	0.380273	0.392081

Continued on next page

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Pea early-browning virus	-	-	-	-	-	-
54	Raspberry leaf mottle virus	0.395843	0.360051	0.393782	0.352979	0.324055	0.360875
	Babaco mosaic virus	0.490517	0.410888	0.411416	0.350772	0.378001	0.364946
	Helenium virus S	-	-	-	-	-	-
55	Peach marafivirus D	0.367533	0.356730	0.323572	0.341199	0.361751	0.338465
	Oat golden stripe virus	0.608279	0.406621	0.339583	0.448225	0.354787	0.320002
	Hydrangea ringspot virus	-	-	-	-	-	-
56	Cherry necrotic rusty mottle virus	0.487347	0.379988	0.343312	0.330284	0.362833	0.564816
	Cordyline virus 1	0.529483	0.429054	0.360698	0.398670	0.383972	0.388333
	Pineapple mealybug wilt-associated virus 1	-	-	-	-	-	-
57	Cucumis melo endornavirus	0.412588	0.324412	0.377488	0.402724	0.303954	0.409605
	Rupestris stem pitting-associated virus	0.440595	0.364207	0.406147	0.425102	0.326103	0.378378
	Lilac ring mottle virus	-	-	-	-	-	-
58	Gentian ovary ring-spot virus	0.310778	0.360135	0.000000	0.364681	0.392145	0.394129
	White clover mosaic virus	0.369623	0.377732	0.416869	0.382905	0.418236	0.385123
	Bamboo mosaic virus	-	-	-	-	-	-
59	Daphne virus S	0.395484	0.364912	0.375297	0.407799	0.351613	0.401197
	Carnation necrotic fleck virus	0.353286	0.458879	0.391517	0.371982	0.405166	0.398109
	Potato virus H	-	-	-	-	-	-
60	Hardenbergia virus A	0.346120	0.330970	0.342788	0.361255	0.369758	0.316752
	Hoya chlorotic spot virus	0.397361	0.344752	0.366615	0.407754	0.369391	0.392149
	Arachis pintoi virus	-	-	-	-	-	-
61	Hippeastrum latent virus	0.352152	0.326047	0.375369	0.428352	0.384518	0.446354
	Lagenaria siceraria endornavirus-Hubei	0.467740	0.358937	0.413828	0.379438	0.398496	0.445197
	Potato yellow vein virus	-	-	-	-	-	-
62	Arracacha virus 1	0.323656	0.366393	0.310586	0.381014	0.375317	0.407957
	Peach chlorotic mottle virus	0.390306	0.403101	0.375509	0.400880	0.375337	0.384071
	Arracacha virus V	-	-	-	-	-	-
63	Actinidia seed-borne latent virus	0.322511	0.352790	0.351670	0.418527	0.349635	0.407066
	Tomato mosaic virus	0.432232	0.389354	0.374736	0.404298	0.426423	0.381134
	Arracacha virus 1	-	-	-	-	-	-
64	Diodia vein chlorosis virus	0.359200	0.410511	0.439089	0.410728	0.350749	0.370119
	Ribes americanum virus A	0.431807	0.380836	0.564567	0.361861	0.390864	0.430716
	Potato aucuba mosaic virus	-	-	-	-	-	-
65	Currant virus A	0.502369	0.373073	0.398768	0.375203	0.375242	0.390600
	Rice stripe necrosis virus	0.529366	0.371868	0.429160	0.412179	0.477545	0.413120
	Tomato mosaic virus	-	-	-	-	-	-
66	Tomato aspermy virus	0.295802	0.396192	0.366172	0.347805	0.111111	10.000000
	Jasmine virus C	0.326087	0.403872	0.394679	0.387047	10.000000	10.000000
	Pepper ringspot virus	-	-	-	-	-	-
67	Oat golden stripe virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Citrus yellow vein clearing virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Hippeastrum latent virus	-	-	-	-	-	-
68	Diuris virus B	10.000000	0.107143	10.000000	1.000000	10.000000	10.000000
	Lychnis ringspot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Peach chlorotic mottle virus	-	-	-	-	-	-
69	Plantago asiatica mosaic virus	10.000000	10.000000	10.000000	1.000000	10.000000	10.000000
	Ribgrass mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Potato aucuba mosaic virus	-	-	-	-	-	-
70	Cactus virus X	0.400000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 13	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Yam latent virus	-	-	-	-	-	-
71	Yam latent virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Clitoria yellow mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Indian peanut clump virus	-	-	-	-	-	-
72	Brome mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Apple chlorotic leaf spot virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Amazon lily mild mottle virus	-	-	-	-	-	-
73	Zucchini green mottle mosaic virus	10.000000	10.000000	1.000000	10.000000	10.000000	10.000000
	Papaya mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 3	-	-	-	-	-	-
74	Odontoglossum ringspot virus	10.000000	10.000000	10.000000	10.000000	10.000000	1.000000
	Grapevine virus A	10.000000	10.000000	10.000000	10.000000	1.000000	10.000000
	Tobacco rattle virus	-	-	-	-	-	-
75	Brome mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cucumber vein-clearing virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Alstroemeria virus x	-	-	-	-	-	-
76	Potato virus M	0.500000	10.000000	10.000000	10.000000	10.000000	10.000000
	Lettuce chlorosis virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	American plum line pattern virus	-	-	-	-	-	-
77	Poa semilatent virus	10.000000	0.000000	10.000000	10.000000	10.000000	10.000000
	Potato virus T	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Youcai mosaic virus	-	-	-	-	-	-
78	Obuda pepper virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Hydrangea chlorotic mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cnidium virus X	-	-	-	-	-	-
79	Lettuce chordovirus 1	0.333333	10.000000	10.000000	10.000000	10.000000	10.000000
	Apple mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Lilac leaf chlorosis virus	-	-	-	-	-	-
80	Carnation necrotic fleck virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Garlic virus X	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Nerine virus X	-	-	-	-	-	-
81	Coleus vein necrosis virus	1.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Turnip vein-clearing virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Passion fruit mosaic virus	-	-	-	-	-	-
82	Bell pepper endornavirus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Garlic mite-borne filamentous virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Blackcurrant leafroll-associated virus 1	-	-	-	-	-	-
83	Prunus necrotic ringspot virus	1.000000	10.000000	0.767346	0.663139	10.000000	10.000000
	Helenium virus S	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Soil-borne cereal mosaic virus	-	-	-	-	-	-
84	Ribgrass mosaic virus	10.000000	1.159770	10.000000	10.000000	0.887698	0.341871
	Clover yellow mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cucurbit chlorotic yellows virus	-	-	-	-	-	-
85	Asparagus virus 2	0.931603	0.738759	10.000000	2.356410	10.000000	0.863917
	Allium virus X	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine leafroll-associated virus 1	-	-	-	-	-	-
86	Pineapple mealybug wilt-associated virus 3	10.000000	1.213820	1.637710	10.000000	0.451688	10.000000
	Sweet potato chlorotic fleck virus	10.000000	10.000000	1.537370	10.000000	10.000000	10.000000
	Heracleum latent virus	-	-	-	-	-	-
87	Peach mosaic virus	10.000000	1.275630	0.868659	10.000000	10.000000	1.931110
	Paprika mild mottle virus	10.000000	10.000000	10.000000	10.000000	2.369560	10.000000
	Little cherry virus 1	-	-	-	-	-	-
88	Tomato necrotic streak virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Grapevine virus F	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Senna mosaic virus	-	-	-	-	-	-
89	Allium virus X	1.801830	0.952700	10.000000	10.000000	10.000000	1.504220
	Burdock mottle virus	1.791760	10.000000	10.000000	10.000000	10.000000	10.000000

Continued on next page

Table D.3 Genus-Family outgroup identification benchmark results

	Comparator Genome In-group Genome Out-group Genome	MoM (in) MoM (out)	BM2 (in) BM2 (out)	Swp (in) Swp (out)	Msh (in) Msh (out)	CoP (in) CoP (out)	SlS (in) SlS (out)
	Euonymus yellow vein virus	-	-	-	-	-	-
90	Scallion virus X	1.020270	10.000000	10.000000	10.000000	10.000000	10.000000
	Rattail cactus necrosis associated virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cucurbit chlorotic yellows virus	-	-	-	-	-	-
91	Zucchini green mottle mosaic virus	10.000000	10.000000	10.000000	1.233940	0.001675	10.000000
	Grapevine virus B	10.000000	10.000000	10.000000	10.000000	1.281450	1.639020
	Olive latent virus 2	-	-	-	-	-	-
92	Rehmannia virus 1	10.000000	0.577178	10.000000	2.076980	0.685055	10.000000
	Grapevine rupestris vein feathering virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Raspberry leaf mottle virus	-	-	-	-	-	-
93	Cucurbit yellow stunting disorder virus	10.000000	10.000000	1.423160	10.000000	1.846300	0.287043
	Strawberry mild yellow edge virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Passion fruit mosaic virus	-	-	-	-	-	-
94	Bean yellow disorder virus	10.000000	10.000000	10.000000	1.489090	1.376070	10.000000
	Lagenaria mild mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Phaseolus vulgaris alphaendornavirus 2	-	-	-	-	-	-
95	Cucumber mosaic virus	10.000000	10.000000	10.000000	2.262370	1.496230	1.297910
	Clover yellow mosaic virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Rubus canadensis virus 1	-	-	-	-	-	-
96	Agave tequilana leaf virus	10.000000	10.000000	10.000000	10.000000	1.690630	1.453220
	Oat golden stripe virus	10.000000	10.000000	10.000000	2.619840	10.000000	0.228694
	Blackberry vein banding associated virus	-	-	-	-	-	-
97	Rose leaf rosette-associated virus	0.933693	10.000000	10.000000	10.000000	0.391035	10.000000
	Atractylodes mottle virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Tea plant line pattern virus	-	-	-	-	-	-
98	Beet yellows virus	10.000000	10.000000	10.000000	10.000000	0.763154	2.789230
	Cherry virus A	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
	Cherry virus A	-	-	-	-	-	-
99	Carrot betaflexivirus 1	10.000000	10.000000	2.720580	10.000000	0.613125	10.000000
	Hibiscus latent Fort Pierce virus	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

Values returned by programs that were larger than 10.0 sub/bp, not-a-number values, or error codes are presumed to indicate a maximum level of divergence, and are represented by 10.0 sub/bp. MoM: Mottle-Map, BM2: BWA-Mem2, Swp: Swipe, Msh: Mash, CoP: Co-Phylog, Sls: Slope-SPAM

References

Adams, I. P., Abad, J., Fribourg, C. E., Boonham, N., and Jones, R. A. (2018). Complete genome sequence of potato virus T from bolivia, obtained from a 33-Year-Old sample. *Microbiology Resource Announcements*, 7(18).

Adams, I. P., Boonham, N., and Jones, R. A. (2017). First complete genome sequence of arracacha virus a isolated from a 38-Year-Old sample from peru. *Genome Announcements*, 5(18).

Afiahayati, K. S. (2015). MetaVelvet-SL: An extension of the velvet assembler to a de novo metagenomic assembler utilizing supervised learning. *DNA Research: An International Journal for Rapid Publication of Reports on Genes and Genomes*, 22(1):69–77.

Afiahayati, K. S. and Sakakibara, Y. (2013). An extended genovo metagenomic assembler by incorporating paired-end information. *PeerJ*, 1:196.

Aganezov, S. S. and Alekseyev, M. A. (2017). CAMSA: A tool for comparative analysis and merging of scaffold assemblies. *BMC Bioinformatics*, 18(Suppl 15):496.

Aiewsakun, P. and Simmonds, P. (2018). The genomic underpinnings of eukaryotic virus taxonomy: Creating a sequence-based framework for family-level virus classification. *Microbiome*, 6(1):38.

Al Rwahnih, M., Daubert, S., Urbez-Torres, J., Cordero, F., and Rowhani, A. (2011). Deep sequencing evidence from single grapevine plants reveals a virome dominated by mycoviruses. *Archives of Virology*, 156(3):397–403.

Albery, G. F., Becker, D. J., Brierley, L., Brook, C. E., Christofferson, R. C., Cohen, L. E., Dallas, T. A., Eskew, E. A., Fagre, A., Farrell, M. J., Glennon, E., Guth, S., Joseph, M. B., Mollentze, N., Neely, B. A., Poisot, T., Rasmussen, A. L., Ryan, S. J., Seifert, S., Sjodin, A. R., Sorrell, E. M., and Carlson, C. J. (2021). The science of the host–virus network. *Nature Microbiology*, 6(12):1483–1492.

Alcalá-Briseño, R. I., Casarrubias-Castillo, K., López-Ley, D., Garrett, K. A., and Silva-Rosales, L. (2020). Network Analysis of the Papaya Orchard Virome from Two Agroecological Regions of Chiapas, Mexico. *mSystems*, 5(1):e00423–19.

Allam, A., Kalnis, P., and Solovyev, V. (2015). Karect: Accurate correction of substitution, insertion and deletion errors for next-Generation sequencing data. *Bioinformatics (Oxford, England)*, 31(21):3421–28.

Alneberg, J., Bjarnason, B. S., Bruijn, I., Schirmer, M., Quick, J., Ijaz, U. Z., Lahti, L., Loman, N. J., Andersson, A. F., and Quince, C. (2014). Binning metagenomic contigs by coverage and composition. *Nature Methods*, 11(11):1144–46.

Alves, J. M., Oliveira, A. L., Sandberg, T. O., Moreno-Gallego, J. L., Toledo, M. A., Moura, E. M., and Oliveira, L. S. (2016). GenSeed-HMM: A tool for progressive assembly using profile HMMs as seeds and its application in alpavirinae viral discovery from metagenomic data.

Amann, R., Ludwig, W., and Schleifer, K. (1995). Phylogenetic identification and in situ detection of individual microbial cells without cultivation. *Microbiological Reviews*, 59(1):143–69.

Amgarten, D., Braga, L. P., Silva, A. M., and Setubal, J. C. (2018). MARVEL, a tool for prediction of bacteriophage sequences in metagenomic bins. *Frontiers in Genetics*, 9(August):304.

Anaconda (2016). Anaconda Software Distribution. https://anaconda.com.

Anderson, N. G., Gerin, J. L., and Anderson, N. (2003). Global screening for human viral pathogens. *Emerging Infectious Diseases*, 9(7):768–74.

Angly, F. E., Felts, B., Breitbart, M., Salamon, P., Edwards, R. A., Carlson, C., and Chan, A. M. (2006). The marine viromes of four oceanic regions. *PLoS Biology*, 4(11):368.

Arumugam Lab (2023). Msamtools: Microbiome-related extension to samtools.

Asplund, M., Kjartansdóttir, K. R., Mollerup, S., Vinner, L., Fridholm, H., Herrera, J. A., and Friis-Nielsen, J. (2019). Contaminating viral sequences in high-throughput sequencing viromics: A linkage study of 700 sequencing libraries. *Clinical Microbiology and Infection: The Official Publication of the European Society of Clinical Microbiology and Infectious Diseases*.

Azmat, M. A., Khan, I. A., Cheema, H. M. N., Rajwana, I. A., Khan, A. S., and Khan, A. A. (2012). Extraction of DNA suitable for PCR applications from mature leaves of mangifera indica L. *Journal of Zhejiang University. Science. B*, 13(4):239–43.

Baaijens, J. A., Aabidine, A. Z. E., Rivals, E., and Schönhuth, A. (2017). De novo assembly of viral quasispecies using overlap graphs. *Genome Research*, 27(5):835–48.

Bagley, S. T. (1985). Habitat Association of Klebsiella Species. *Infection Control & Hospital Epidemiology*, 6(2):52–58.

Bailey, T. L., Johnson, J., Grant, C. E., and Noble, W. S. (2015). The MEME suite. *Nucleic Acids Research*, 43(W1):39–49.

Baker, D. N. and Langmead, B. (2019). Dashing: Fast and accurate genomic distances with HyperLogLog. *Genome Biology*, 20(1):265.

Bao, E. and Lan, L. (2017). HALC: High throughput algorithm for long read error correction. *BMC Bioinformatics*, 18(1):204.

Bao, E., Xie, F., Song, C., and Song, D. (2019). FLAS: Fast and high throughput algorithm for PacBio long read self-correction. *Bioinformatics (Oxford, England)*.

Barba, M., Czosnek, H., and Hadidi, A. (2014). Historical perspective, development and applications of next-Generation sequencing in plant virology. *Viruses*, 6(1):106–36.

Barrientos-Somarribas, M., Messina, D. N., Pou, C., Lysholm, F., Bjerkner, A., Allander, T., Andersson, B., and Sonnhammer, E. L. L. (2018). Discovering viral genomes in human metagenomic data by predicting unknown protein families. *Scientific Reports*, 8(1):28.

Barrio-Hernandez, I., Yeo, J., Jänes, J., Mirdita, M., Gilchrist, C. L. M., Wein, T., Varadi, M., Velankar, S., Beltrao, P., and Steinegger, M. (2023). Clustering predicted structures at the scale of the known protein universe. *Nature*, 622(7983):637–645.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Sayers, E. W. (2011). GenBank. *Nucleic Acids Research*, 39.

Bertels, F., Silander, O. K., Pachkov, M., Rainey, P. B., and van Nimwegen, E. (2014). Automated Reconstruction of Whole-Genome Phylogenies from Short-Sequence Reads. *Molecular Biology and Evolution*, 31(5):1077–1088.

Bester, R., Cook, G., Breytenbach, J. H. J., Steyn, C., De Bruyn, R., and Maree, H. J. (2021). Towards the validation of high-throughput sequencing (HTS) for routine plant virus diagnostics: Measurement of variation linked to HTS detection of citrus viruses and viroids. *Virology Journal*, 18:61.

Blanc, S., Uzest, M., and Drucker, M. (2011). New research horizons in vector-transmission of plant viruses. *Current Opinion in Microbiology*, 14(4):483–91.

Bodily, P. M., Fujimoto, M., Snell, Q., Ventura, D., and Clement, M. J. (2016). ScaffoldScaffolder: Solving contig orientation via bidirected to directed graph reduction. *Bioinformatics (Oxford, England)*, 32(1):17–24.

Boetzer, M. and Pirovano, W. (2014). SSPACE-LongRead: Scaffolding bacterial draft genomes using long read sequence information. *BMC Bioinformatics*, 15(June):211.

Boisvert, S., Raymond, F., Godzaridis, E., Laviolette, F., and Corbeil, J. (2012). Ray meta: Scalable de novo metagenome assembly and profiling. *Genome Biology*, 13(12):122.

Bolduc, B., Shaughnessy, D. P., Wolf, Y. I., Koonin, E. V., Roberto, F. F., and Young, M. (2012). Identification of novel positive-strand RNA viruses by metagenomic analysis of archaea-dominated yellowstone hot springs. *Journal of Virology*, 86(10):5562–73.

Bolger, A. M., Lohse, M., and Usadel, B. (2014). Trimmomatic: A Flexible Trimmer for Illumina Sequence Data. *Bioinformatics (Oxford, England)*, 30(15):2114–20.

Bosch, A., Gkogka, E., Le Guyader, F. S., Loisy-Hamon, F., Lee, A., van Lieshout, L., Marthi, B., Myrmel, M., Sansom, A., Schultz, A. C., Winkler, A., Zuber, S., and Phister, T. (2018). Foodborne viruses: Detection, risk assessment, and control options in food processing. *International Journal of Food Microbiology*, 285:110–128.

Bosi, E., Donati, B., Galardini, M., Brunetti, S., Sagot, M.-F., Lió, P., Crescenzi, P., Fani, R., and Fondi, M. (2015). MeDuSa: A multi-draft based scaffolder. *Bioinformatics (Oxford, England)*, 31(15):2443–51.

Brault, V., Uzest, M., Monsion, B., Jacquot, E., and Blanc, S. (2010). Aphids as transport devices for plant viruses. *Comptes Rendus Biologies*, 333(6-7):524–38.

Breitbart, M., Delwart, E., Rosario, K., Segalés, J., Varsani, A., and ICTV Report Consortium (2017). ICTV Virus Taxonomy Profile: Circoviridae. *Journal of General Virology*, 98(8):1997–1998.

Breitbart, M., Hewson, I., Felts, B., Mahaffy, J. M., Nulton, J., Salamon, P., and Rohwer, F. (2003). Metagenomic analyses of an uncultured viral community from human feces. *Journal of Bacteriology*, 185(20):6220–23.

Breitbart, M., Salamon, P., Andresen, B., Mahaffy, J. M., Segall, A. M., Mead, D., Azam, F., and Rohwer, F. (2002). Genomic analysis of uncultured marine viral communities. *Proceedings of the National Academy of Sciences of the United States of America*, 99(22):14250–55.

Brenner, S., Johnson, M., Bridgham, J., Golda, G., Lloyd, D., Johnson, D., and Luo, S. (2000). Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays. *Nature Biotechnology*, 18(6):630–34.

Brister, J. R., Ako-Adjei, D., Bao, Y., and Blinkova, O. (2015). NCBI viral genomes resource. *Nucleic Acids Research*, 43(Database issue):D571–577.

Buchfink, B., Xie, C., and Huson, D. H. (2015). Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, 12(1):59–60.

Bushmanova, E., Antipov, D., Lapidus, A., and Prjibelski, A. D. (2018). rnaSPAdes: A de novo transcriptome assembler and its application to RNA-Seq data.

Bzhalava, Z., Tampuu, A., Bała, P., Vicente, R., and Dillner, J. (2018). Machine learning for detection of viral sequences in human metagenomic datasets. *BMC Bioinformatics*, 19(1):336.

CALIBER (2023). CALIBER – Bacterial Plant Diseases Programme. https://bacterialplantdiseases.uk/caliber/.

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., and Madden, T. L. (2009). BLAST+: Architecture and applications. *BMC Bioinformatics*, 10(December):421.

Cao, M., Zhang, S., Liao, R., Wang, X., Xuan, Z., Zhan, B., Li, Z., Zhang, J., Du, X., Tang, Z., Li, S., and Zhou, Y. (2021). Spatial Virome Analysis of Zanthoxylum armatum Trees Affected With the Flower Yellowing Disease. *Frontiers in Microbiology*, 12:702210.

Card, S., Pearson, M., and Clover, G. (2007). Plant pathogens transmitted by pollen. *Australasian Plant Pathology: APP*, 36(5):455–61.

Carroll, D., Morzaria, S., Briand, S., Johnson, C. K., Morens, D., Sumption, K., Tomori, O., and Wacharphaueasadee, S. (2021). Preventing the next pandemic: The power of a global viral surveillance network. *BMJ*, 372:n485.

Center for Algorithmic Biotechnology (2022). Graph construction. https://github.com/ablab/spades/wiki/Graph-construction.

Chang, Z., Li, G., Liu, J., Zhang, Y., Ashby, C., Liu, D., Cramer, C. L., and Huang, X. (2015). Bridger: A new framework for de novo transcriptome assembly using RNA-Seq data. *Genome Biology*, 16(February):30.

Chen, C., Khaleel, S. S., Huang, H., and Wu, C. H. (2014). Software for pre-processing illumina next-Generation sequencing short read sequences. *Source Code for Biology and Medicine*, 9.

Chen, K.-T., Chen, C.-J., Shen, H.-T., Liu, C.-L., Huang, S.-H., and Lu, C. L. (2016). Multi-CAR: A tool of contig scaffolding using multiple references. *BMC Bioinformatics*, 17(Suppl 17):469.

Chen, K.-T., Shen, H.-T., and Lu, C. L. (2018a). Multi-CSAR: A multiple reference-based contig scaffolder using algebraic rearrangements. *BMC Systems Biology*, 12(Suppl 9):139.

Chen, S., Zhou, Y., Chen, Y., and Gu, J. (2018b). Fastp: An ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics (Oxford, England)*, 34(17):i884–i890.

Chen, Y.-M., Yu, C.-H., Hwang, C.-C., and Liu, T. (2013). OMACC: An optical-map-assisted contig connector for improving de novo genome assembly. *BMC Systems Biology*, 7 Suppl 6.

Chibani, C. M., Farr, A., Klama, S., Dietrich, S., and Liesegang, H. (2019). Classifying the unclassified: A phage classification method. *Viruses*, 11(2).

Clem, A. L., Sims, J., Telang, S., Eaton, J. W., and Chesney, J. (2007). Virus detection and identification using random multiplex (RT)-PCR with 3'-locked random primers. *Virology Journal*, 4:65.

Coetzee, B., Freeborough, M.-J., Maree, H. J., Jean-Marc Celton, D. G., and Burger, J. T. (2010). Deep sequencing analysis of viruses infecting grapevines: Virome of a vineyard. *Virology*, 400(2):157–63.

Criscuolo, A. (2020). On the transformation of MinHash-based uncorrected distances into proper evolutionary distances for phylogenetic inference. *F1000Research*, 9:1309.

Criscuolo, A. and Brisse, S. (2013). AlienTrimmer: A tool to quickly and accurately trim off multiple short contaminant sequences from high-throughput sequencing reads. *Genomics*, 102(5-6):500–506.

Dáder, B., Then, C., Berthelot, E., Ducousso, M., Ng, J. C., and Drucker, M. (2017). Insect transmission of plant viruses: Multilayered interactions optimize viral propagation. *Insect Science*, 24(6):929–46.

Dayarian, A., Michael, T. P., and Sengupta, A. M. (2010). SOPRA: Scaffolding algorithm for paired reads via statistical optimization. *BMC Bioinformatics*, 11(June):345.

de Vries, J. J., Brown, J. R., Fischer, N., Sidorov, I. A., Morfopoulou, S., Huang, J., Munnink, B. B. O., Sayiner, A., Bulgurcu, A., Rodriguez, C., Gricourt, G., Keyaerts, E., Beller, L., Bachofen, C., Kubacki, J., Cordey, S., Laubscher, F., Schmitz, D., Beer, M., Hoeper, D., Huber, M., Kufner, V., Zaheri, M., Lebrand, A., Papa, A., van Boheemen, S., Kroes, A. C., Breuer, J., Lopez-Labrador, F. X., and Claas, E. C. (2021). Benchmark of thirteen bioinformatic pipelines for metagenomic virus diagnostics using datasets from clinical samples. *Journal of clinical virology : the official publication of the Pan American Society for Clinical Virology*, 141:104908.

Desbiez, C., Moury, B., and Lecoq, H. (2011). The hallmarks of 'green' viruses: Do plant viruses evolve differently from the others?" infection. *Genetics and Evolution: Journal of Molecular Epidemiology and Evolutionary Genetics in Infectious Diseases*, 11(5):812–24.

Didion, J. P., Martin, M., and Collins, F. S. (2017). Atropos: Specific, sensitive, and speedy trimming of sequencing reads. *PeerJ*, 5.

Dietzgen, R. G., Mann, K. S., and Johnson, K. N. (2016). Plant virus-insect vector interactions: Current and potential future research directions. *Viruses*, 8(11).

Dominguez-Huerta, G., Wainaina, J. M., Zayed, A. A., Culley, A. I., Kuhn, J. H., and Sullivan, M. B. (2023). The RNA virosphere: How big and diverse is it? *Environmental Microbiology*, 25(1):209–215.

Donmez, N. and Brudno, M. (2013). SCARPA: Scaffolding reads with practical algorithms. *Bioinformatics (Oxford, England)*, 29(4):428–34.

Dou, J., Dou, H., Mu, C., Zhang, L., Li, Y., Wang, J., and Li, T. (2017). Whole-genome restriction mapping by 'Subhaploid'-Based RAD sequencing: An efficient and flexible approach for physical mapping and genome scaffolding.

Douglass, A. P., O'Brien, C. E., Offei, B., Coughlan, A. Y., Ortiz-Merino, R. A., Butler, G., Byrne, K. P., and Wolfe, K. H. (2019). Coverage-Versus-Length Plots, a Simple Quality Control Step for de Novo Yeast Genome Sequence Assemblies. *G3: Genes|Genomes|Genetics*, 9(3):879–887.

Duffy, S., Shackelton, L. A., and Holmes, E. C. (2008). Rates of evolutionary change in viruses: Patterns and determinants. *Nature Reviews Genetics*, 9(4):267–276.

Durai, D. A. and Schulz, M. H. (2019). Improving in-Silico normalization using read weights. *Scientific Reports*, 9(1):5133.

Eddy, S. R. (2011). Accelerated profile HMM searches. *PLoS Computational Biology*, 7(10):1002195.

Edwards, R. A. and Rohwer, F. (2005). Viral Metagenomics. *Nature Reviews. Microbiology*, 3(6):504–10.

Eigner, J., Boedtker, H., and Michaels, G. (1961). The thermal degradation of nucleic acids. *Biochimica et Biophysica Acta*, 51(July):165–68.

Falgueras, J., Lara, A. J., Fernández-Pozo, N., Cantón, F. R., Pérez-Trabado, G., and Claros, M. (2010). SeqTrim: A high-throughput pipeline for pre-processing any type of sequence read. *BMC Bioinformatics*, 11(January):38.

Farrant, G. K., Hoebeke, M., Partensky, F., Andres, G., Corre, E., and Garczarek, L. (2015). WiseScaffolder: An algorithm for the semi-automatic scaffolding of next generation sequencing data. *BMC Bioinformatics*, 16(September):281.

Federhen, S. (2012). The NCBI Taxonomy database. *Nucleic Acids Research*, 40(D1):D136–D143.

Felsenstein, J. (2022). The Newick tree format. https://phylipweb.github.io/phylip/newicktree.html.

Fierer, N., Breitbart, M., Nulton, J., Salamon, P., Lozupone, C., Jones, R., and Robeson, M. (2007). Metagenomic and small-subunit rRNA analyses reveal the genetic diversity of bacteria, archaea, fungi, and viruses in soil. *Applied and Environmental Microbiology*, 73(21):7059–66.

Flamholz, Z. N., Biller, S. J., and Kelly, L. (2023). Large language models improve annotation of viral proteins. *Research Square*, pages rs.3.rs–2852098.

Fondi, M., Orlandini, V., Corti, G., Severgnini, M., Galardini, M., Pietrelli, A., and Fuligni, F. (2014). Enly: Improving draft genomes through reads recycling. *Journal of Genomics*, 2(April):89–93.

Fowkes, A. R., McGreig, S., Pufal, H., Duffy, S., Howard, B., Adams, I. P., Macarthur, R., Weekes, R., and Fox, A. (2021). Integrating High throughput Sequencing into Survey Design Reveals Turnip Yellows Virus and Soybean Dwarf Virus in Pea (Pisum Sativum) in the United Kingdom. *Viruses*, 13(12):2530.

Gaafar, Y. Z. A. and Ziebell, H. (2020). Comparative study on three viral enrichment approaches based on RNA extraction for plant virus/viroid detection using high-throughput sequencing. *PloS One*, 15(8):e0237951.

Gao, S., Bertrand, D., Chia, B. K., and Nagarajan, N. (2016). OPERA-LG: Efficient and exact scaffolding of large, repeat-rich eukaryotic genomes with performance guarantees. *Genome Biology*, 17(May).

Gao, S., Sung, W.-K., and Nagarajan, N. (2011). Opera: Reconstructing optimal genomic scaffolds with high-throughput paired-end sequences. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 18(11):1681–91.

García-López, R., Vázquez-Castellanos, J. F., and Moya, A. (2015). Fragmentation and Coverage Variation in Viral Metagenome Assemblies, and Their Effect in Diversity Calculations. *Frontiers in Bioengineering and Biotechnology*, 3:141.

Girgis, H. Z., James, B. T., and Luczak, B. B. (2021). Identity: Rapid alignment-free prediction of sequence alignment identity scores using self-supervised general linear models. *NAR genomics and bioinformatics*, 3(1):lqab001.

Goodacre, N., Aljanahi, A., Nandakumar, S., Mikailov, M., and Khan, A. S. (2018). A Reference Viral Database (RVDB) To Enhance Bioinformatics Analysis of High-Throughput Sequencing for Novel Virus Detection. *mSphere*, 3(2):e00069–18.

Gould, E. (1999). Methods for long-term virus preservation. *Molecular Biotechnology*, 13(1):57–66.

Gourlé, H., Karlsson-Lindsjö, O., Hayer, J., and Bongcam-Rudloff, E. (2019). Simulating Illumina metagenomic data with InSilicoSeq. *Bioinformatics*, 35(3):521–522.

Grabherr, M. G., Haas, B. J., Yassour, M., Levin, J. Z., Thompson, D. A., Amit, I., and Adiconis, X. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature Biotechnology*, 29(7):644–52.

Grabherr, M. G., Russell, P., Meyer, M., Mauceli, E., Alföldi, J., Di Palma, F., and Lindblad-Toh, K. (2010). Genome-wide synteny through highly sensitive sequence alignment: *Satsuma*. *Bioinformatics*, 26(9):1145–1151.

Greenfield, P., Duesing, K., Papanicolaou, A., and Bauer, D. C. (2014). Blue: Correcting Sequencing Errors Using Consensus and Context. *Bioinformatics (Oxford, England)*, 30(19):2723–32.

Gritsenko, A. A., Nijkamp, J. F., Reinders, M. J., and Ridder, D. (2012). GRASS: A generic algorithm for scaffolding next-Generation sequencing assemblies. *Bioinformatics (Oxford, England)*, 28(11):1429–37.

Grubaugh, N. D., Ladner, J. T., Lemey, P., Pybus, O. G., Rambaut, A., Holmes, E. C., and Andersen, K. G. (2019). Tracking virus outbreaks in the twenty-first century. *Nature Microbiology*, 4(1):10–19.

Habteselassie, M. Y., Bischoff, M., Applegate, B., Reuhs, B., and Turco, R. F. (2010). Understanding the role of agricultural practices in the potential colonization and contamination by Escherichia coli in the rhizospheres of fresh produce. *Journal of Food Protection*, 73(11):2001–2009.

Hachiya, T., Osana, Y., Popendorf, K., and Sakakibara, Y. (2009). Accurate identification of orthologous segments among multiple genomes. *Bioinformatics*, 25(7):853–860.

Haegeman, A. (2021). ILVO / VIROMOCKchallenge · GitLab. https://gitlab.com/ilvo/VIROMOCKchallenge.

Haider, B., Ahn, T.-H., Bushnell, B., Chai, J., Copeland, A., and Pan, C. (2014). Omega: An Overlap-Graph de Novo Assembler for Metagenomics. *Bioinformatics (Oxford, England)*, 30(19):2717–22.

Handelsman, J., Rondon, M., Brady, S., Clardy, J., and Goodman, R. (1998). Molecular biological access to the chemistry of unknown soil microbes: A new frontier for natural products. *Chemistry & Biology*, 5(10):245–49.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Van Kerkwijk, M. H., Brett, M., Haldane, A., Del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825):357–362.

Haubold, B., Pfaffelhuber, P., Domazet-Los ̆o, M., and Wiehe, T. (2009). Estimating Mutation Distances from Unaligned Genomes. *Journal of Computational Biology*, 16(10):1487–1500.

Hayden, E. C. (2014). Technology: The $1,000 genome. *Nature*, 507(7492):294–295.

Herath, D., Tang, S.-L., Tandon, K., Ackland, D., and Halgamuge, S. K. (2017). CoMet: A workflow using contig coverage and composition for binning a metagenomic sample with high precision. *BMC Bioinformatics*, 18(Suppl 16):571.

Ho, T. and Tzanetakis, I. E. (2014). Development of a virus detection and discovery pipeline using next generation sequencing. *Virology*, 471-473 (December:54–60.

Holland, J., Spindler, K., Horodyski, F., Grabau, E., Nichol, S., and VandePol, S. (1982). Rapid Evolution of RNA Genomes. *Science (New York, N.Y.)*, 215(4540):1577–85.

Houldcroft, C. J., Beale, M. A., and Breuer, J. (2017). Clinical and biological insights from viral genome sequencing. *Nature Reviews. Microbiology*, 15(3):183–92.

Hounsome, L., Herr, D., Bryant, R., Smith, R., Loman, L., Harris, J., Youhan, U., Dzene, E., Hadjipantelis, P., Long, H., Laurence, T., Riley, S., and Cumming, F. (2022). Epidemiological impact of a large number of incorrect negative SARS-CoV-2 test results in South West England during September and October 2021.

Hsiung, G. D. (1984). Diagnostic virology: From animals to automation. *The Yale Journal of Biology and Medicine*, 57(5):727–733.

Huang, Y.-T. and Huang, Y.-W. (2017). An efficient error correction algorithm using FM-Index. *BMC Bioinformatics*, 18(1):524.

Huerta-Cepas, J., Serra, F., and Bork, P. (2016). ETE 3: Reconstruction, Analysis, and Visualization of Phylogenomic Data. *Molecular Biology and Evolution*, 33(6):1635–1638.

Hugenholtz, P., Goebel, B., and Pace, N. (1998). Impact of culture-independent studies on the emerging phylogenetic view of bacterial diversity. *Journal of Bacteriology*, 180(18):4765–74.

Hunt, M., Gall, A., Ong, S. H., Brener, J., Ferns, B., Goulder, P., Nastouli, E., Keane, J. A., Kellam, P., and Otto, T. D. (2015). IVA: Accurate de Novo Assembly of RNA Virus Genomes. *Bioinformatics (Oxford, England)*, 31(14):2374–76.

Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9(3):90–95.

Hurwitz, B. L. and Sullivan, M. B. (2013). The Pacific Ocean virome (POV): A marine viral metagenomic dataset and associated protein clusters for quantitative viral ecology. *PloS One*, 8(2):e57355.

Huson, D. H., Tappu, R., Bazinet, A. L., Xie, C., Cummings, M. P., Nieselt, K., and Williams, R. (2017). Fast and simple protein-alignment-guided assembly of orthologous gene families from microbiome sequencing reads. *Microbiome*, 5(1):11.

Illumina (2023). TruSeq Stranded Total RNA with Ribo-Zero Plant | For plant transcriptome studies. https://emea.illumina.com/products/by-type/sequencing-kits/library-prep-kits/truseq-stranded-total-rna-plant.html.

Ilyas, R., Rohde, M. J., Richert-Pöggeler, K. R., and Ziebell, H. (2022). To Be Seen or Not to Be Seen: Latent Infection by Tobamoviruses. *Plants (Basel, Switzerland)*, 11(16):2166.

Imelfort, M., Parks, D., Woodcroft, B. J., Dennis, P., Hugenholtz, P., and Tyson, G. W. (2014). GroopM: An automated tool for the recovery of population genomes from related metagenomes. *PeerJ*, 2:603.

Iqbal, Z., Caccamo, M., Turner, I., Flicek, P., and McVean, G. (2012). De novo assembly and genotyping of variants using colored de bruijn graphs. *Nature Genetics*, 44(2):226–32.

Jackman, S. D., Coombe, L., Chu, J., Warren, R. L., Vandervalk, B. P., Yeo, S., and Xue, Z. (2018). Tigmint: Correcting assembly errors using linked reads from large molecules. *BMC Bioinformatics*, 19(1):393.

Jackman, S. D., Vandervalk, B. P., Mohamadi, H., Chu, J., Sarah Yeo, S. H., and Jahesh, G. (2017). ABySS 2.0: Resource-efficient assembly of large genomes using a bloom filter. *Genome Research*, 27(5):768–77.

Jia, D., Chen, Q., Mao, Q., Zhang, X., Wu, W., Chen, H., Yu, X., Wang, Z., and Wei, T. (2018). Vector mediated transmission of persistently transmitted plant viruses. *Current Opinion in Virology*, 28(February):127–32.

Jo, Y., Choi, H., Kim, S.-M., Kim, S.-L., Lee, B. C., and Cho, W. K. (2017). The pepper virome: Natural co-infection of diverse viruses and their quasispecies. *BMC Genomics*, 18(1):453.

Joshi, N. and N, F. J. (2011). Sickle: A sliding-window, adaptive, quality-based trimming tool for FastQ files.

Jukes, T. H. and Cantor, C. R. (1969). Evolution of Protein Molecules. In *Mammalian Protein Metabolism*, pages 21–132. Elsevier.

Kalvari, I., Nawrocki, E. P., Ontiveros-Palacios, N., Argasinska, J., Lamkiewicz, K., Marz, M., Griffiths-Jones, S., Toffano-Nioche, C., Gautheret, D., Weinberg, Z., Rivas, E., Eddy, S. R., Finn, R. D., Bateman, A., and Petrov, A. I. (2021). Rfam 14: Expanded coverage of metagenomic, viral and microRNA families. *Nucleic Acids Research*, 49(D1):D192–D200.

Kang, D. D., Froula, J., Egan, R., and Wang, Z. (2015). MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ*, 3:1165.

Kannan, S., Hui, J., Mazooji, K., Pachter, L., and Tse, D. (2016). Shannon: An Information-Optimal de Novo RNA-Seq Assembler.

Katoh, K., Rozewicki, J., and Yamada, K. D. (2019). MAFFT online service: Multiple sequence alignment, interactive sequence choice and visualization. *Briefings in Bioinformatics*, 20(4):1160–1166.

Kechin, A., Boyarskikh, U., Kel, A., and Filipenko, M. (2017). cutPrimers: A new tool for accurate cutting of primers from reads of targeted next generation sequencing. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 24(11):1138–43.

Kelley, D. R., Schatz, M. C., and Salzberg, S. L. (2010). Quake: Quality-aware detection and correction of sequencing errors. *Genome Biology*, 11(11):116.

Kim, K.-H. and Bae, J.-W. (2011). Amplification methods bias metagenomic libraries of uncultured single-stranded and double-stranded DNA viruses. *Applied and Environmental Microbiology*, 77(21):7663–68.

Klötzl, F. and Haubold, B. (2020). Phylonium: Fast estimation of evolutionary distances from large samples of similar genomes. *Bioinformatics*, 36(7):2040–2046.

Kolmogorov, M., Raney, B., Paten, B., and Pham, S. (2014). Ragout-a reference-assisted assembly tool for bacterial genomes. *Bioinformatics (Oxford, England)*, 30(12):302–9.

Kong, Y. (2011). Btrim: A fast, lightweight adapter and quality trimming program for next-Generation sequencing technologies. *Genomics*, 98(2):152–53.

Koonin, E. V., Krupovic, M., and Dolja, V. V. (2023). The global virome: How much diversity and how many independent origins? *Environmental Microbiology*, 25(1):40–44.

Koren, S., Treangen, T. J., Hill, C. M., Pop, M., and Phillippy, A. M. (2014). Automated ensemble assembly and validation of microbial genomes. *BMC Bioinformatics*, 15.

Koren, S., Treangen, T. J., and Pop, M. (2011). Bambus 2: Scaffolding metagenomes. *Bioinformatics (Oxford, England)*, 27(21):2964–71.

Kosugi, S., Hirakawa, H., and Tabata, S. (2015). GMcloser: Closing gaps in assemblies accurately with a likelihood-based selection of contig or long-read alignments. *Bioinformatics (Oxford, England)*, 31(23):3733–41.

Kreuze, J. F., Perez, A., Untiveros, M., Quispe, D., Fuentes, S., Barker, I., and Simon, R. (2009). Complete viral genome sequence and discovery of novel viruses by deep sequencing of small RNAs: A generic method for diagnosis, discovery and sequencing of viruses. *Virology*, 388(1):1–7.

Krishnamurthy, S. R. and Wang, D. (2017). Origins and challenges of viral dark matter. *Virus Research*, 239:136–142.

Kumar, A., Murthy, S., and Kapoor, A. (2017). Evolution of selective-sequencing approaches for virus discovery and virome analysis. *Virus Research*, 239(July):172–79.

Lai, B., Ding, R., Li, Y., Duan, L., and Zhu, H. (2012). A de Novo Metagenomic Assembly Program for Shotgun DNA Reads. *Bioinformatics (Oxford, England)*, 28(11):1455–62.

Lai, B., Wang, F., Wang, X., Duan, L., and Zhu, H. (2015). InteMAP: Integrated metagenomic assembly pipeline for NGS short reads. *BMC Bioinformatics*, 16(August):244.

Lam, K.-K., Hall, R., Clum, A., and Rao, S. (2016). BIGMAC : Breaking inaccurate genomes and merging assembled contigs for long read metagenomic assembly. *BMC Bioinformatics*, 17(1):435.

Lane, D., Stahl, D., Olsen, G., Heller, D., and Pace, N. (1985). Phylogenetic analysis of the genera thiobacillus and thiomicrospira by 5S rRNA sequences. *Journal of Bacteriology*, 163(1):75–81.

Lane, L. C. (1986). Propagation and purification of RNA plant viruses. In *Methods in Enzymology, 118:687–96*. Academic Press.

Langmead, B., Wilks, C., Antonescu, V., and Charles, R. (2019). Scaling read aligners to hundreds of threads on general-purpose processors. *Bioinformatics*, 35(3):421–432.

Larkin, M. (2003). Human virome project underway. *The Lancet Infectious Diseases*, 3(8):460.

Laserson, J., Jojic, V., and Koller, D. (2011). Genovo: De novo assembly for metagenomes. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 18(3):429–43.

LaTourrette, K., Holste, N. M., and Garcia-Ruiz, H. (2021). Polerovirus genomic variation. *Virus Evolution*, 7(2):veab102.

Lauring, A. S. and Andino, R. (2010). Quasispecies theory and the behavior of RNA viruses. *PLoS Pathogens*, 6(7):1001005.

Le, H.-S., Schulz, M. H., McCauley, B. M., Hinman, V. F., and Bar-Joseph, Z. (2013). Probabilistic error correction for RNA sequencing. *Nucleic Acids Research*, 41(10):109.

Lefkowitz, E. J., Dempsey, D. M., Hendrickson, R. C., Orton, R. J., Siddell, S. G., and Smith, D. B. (2018). Virus taxonomy: The database of the International Committee on Taxonomy of Viruses (ICTV). *Nucleic Acids Research*, 46(D1):D708–D717.

Leimeister, C.-A., Dencker, T., and Morgenstern, B. (2019a). Accurate multiple alignment of distantly related genome sequences using filtered spaced word matches as anchor points. *Bioinformatics*, 35(2):211–218.

Leimeister, C.-A. and Morgenstern, B. (2014). Kmacs: The k -mismatch average common substring approach to alignment-free sequence comparison. *Bioinformatics*, 30(14):2000–2008.

Leimeister, C.-A., Schellhorn, J., Dörrer, S., Gerth, M., Bleidorn, C., and Morgenstern, B. (2019b). Prot-SpaM: Fast alignment-free phylogeny reconstruction based on whole-proteome sequences. *GigaScience*, 8(3).

Leung, H. C., Yiu, S.-M., and Chin, F. Y. (2015). IDBA-MTP: A hybrid metatranscriptomic assembler based on protein information. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 22(5):367–76.

Leung, H. C., Yiu, S.-M., Parkinson, J., and Chin, F. Y. (2013). IDBA-MT: De novo assembler for metatranscriptomic data generated from next-Generation sequencing technology. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 20(7):540–50.

Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R., and Pfister, H. (2014). UpSet: Visualization of Intersecting Sets. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1983–1992.

Li, D., Huang, Y., Leung, C.-M., Luo, R., Ting, H.-F., and Lam, T.-W. (2017). MegaGTA: A sensitive and accurate metagenomic gene-targeted assembler using iterative de bruijn graphs. *BMC Bioinformatics*, 18(Suppl 12):408.

Li, D., Luo, R., Liu, C.-M., Leung, C.-M., Ting, H.-F., Sadakane, K., Yamashita, H., and Lam, T.-W. (2016). MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices. *Methods (San Diego, Calif.)*, 102(June):3–11.

Li, H. (2015). BFC: Correcting Illumina Sequencing Errors. *Bioinformatics (Oxford, England)*, 31(17):2885–87.

Li, R., Zhu, H., Ruan, J., Qian, W., Fang, X., Shi, Z., and Li, Y. (2010). De novo assembly of human genomes with massively parallel short read sequencing. *Genome Research*, 20(2):265–72.

Li, Y. I. and Copley, R. R. (2013). Scaffolding low quality genomes using orthologous protein sequences. *Bioinformatics (Oxford, England)*, 29(2):160–65.

Li, Y.-L., Weng, J.-C., Hsiao, C.-C., Chou, M.-T., Tseng, C.-W., and Hung, J.-H. (2015). PEAT: An Intelligent and Efficient Paired-End Sequencing Adapter Trimming Algorithm. *BMC Bioinformatics*, 16 Suppl 1.

Lin, H.-H. and Liao, Y.-C. (2016). Accurate binning of metagenomic contigs via automated clustering sequences using information of genomic signatures and marker genes. Technical report.

Lin, S.-H. and Liao, Y.-C. (2013). CISA: Contig Integrator for Sequence Assembly of Bacterial Genomes. *PloS One*, 8(3):60843.

Lindsay, J., Salooti, H., Măndoiu, I., and Zelikovsky, A. (2014). ILP-Based Maximum Likelihood Genome Scaffolding. *BMC Bioinformatics*, 15 Suppl 9.

Liu, B., Liu, C.-M., Li, D., Li, Y., Ting, H.-F., Yiu, S.-M., Luo, R., and Lam, T.-W. (2016a). BASE: A practical de novo assembler for large genomes using long NGS reads. *BMC Genomics*, 5(August):499.

Liu, F., Miao, Y., Liu, Y., and Hou, T. (2022). RNN-VirSeeker: A Deep Learning Method for Identification of Short Viral Sequences From Metagenomes. *IEEE/ACM transactions on computational biology and bioinformatics*, 19(3):1840–1849.

Liu, J., Li, G., Chang, Z., Yu, T., Liu, B., McMullen, R., Chen, P., and Huang, X. (2016b). BinPacker: Packing-based de novo transcriptome assembly from RNA-Seq data. *PLoS Computational Biology*, 12(2):1004772.

Liu, X., Yu, Y., Liu, J., Elliott, C. F., Qian, C., and Liu, J. (2018). A novel data structure to support ultra-fast taxonomic classification of metagenomic sequences with K-mer signatures. *Bioinformatics (Oxford, England)*, 34(1):171–78.

Liu, Y., Hou, T., Kang, B., and Liu, F. (2017). Unsupervised binning of metagenomic assembled contigs using improved fuzzy C-means method. *IEEE/ACM Transactions on Computational Biology and Bioinformatics / IEEE, ACM*, 14(6):1459–67.

Lorenzi, H. A., Hoover, J., Inman, J., Safford, T., Murphy, S., Kagan, L., and Williamson, S. J. (2011). TheViral MetaGenome annotation Pipeline(VMGAP):an automated tool for the functional annotation of viral metagenomic shotgun sequencing data. *Standards in Genomic Sciences*, 4(3):418–29.

Lu, R.-B., Lan, P.-X., Kang, R.-J., Tan, G.-L., Chen, X.-J., Li, R., and Li, F. (2022). Genomic characterization of a new enamovirus infecting common bean. *Archives of Virology*, 167(3):999–1002.

Lu, Y. Y., Chen, T., Fuhrman, J. A., and Sun, F. (2017). COCACOLA: Binning metagenomic contigs using sequence COmposition, read CoverAge, CO-Alignment and paired-end read LinkAge. *Bioinformatics (Oxford, England)*, 33(6):791–98.

Luo, J., Wang, J., Zhang, Z., Li, M., and Wu, F.-X. (2017). BOSS: A novel scaffolding algorithm based on an optimized scaffold graph. *Bioinformatics (Oxford, England)*, 33(2):169–76.

Luo, R., Liu, B., Xie, Y., Li, Z., Huang, W., Yuan, J., and He, G. (2012). SOAPdenovo2: An empirically improved memory-efficient short-read de novo assembler. *GigaScience*, 1(1):18.

Maarala, A. I., Bzhalava, Z., Dillner, J., Heljanko, K., and Bzhalava, D. (2018). ViraPipe: Scalable parallel pipeline for viral metagenome analysis from next generation sequencing reads. *Bioinformatics (Oxford, England)*, 34(6):928–35.

Madoui, M.-A., Dossat, C., d'Agata, L., Oeveren, J., Vossen, E., and Aury, J.-M. (2016). MaGuS: A tool for quality assessment and scaffolding of genome assemblies with whole genome ProfilingTM data. *BMC Bioinformatics*, 17.

Madoui, M.-A., Engelen, S., Cruaud, C., Belser, C., Bertrand, L., Alberti, A., Lemainque, A., Wincker, P., and Aury, J.-M. (2015). Genome assembly using nanopore-guided long and error-free DNA reads. *BMC Genomics*, 16.

Mandric, I., Knyazev, S., and Zelikovsky, A. (2018). Repeat-aware evaluation of scaffolding tools. *Bioinformatics (Oxford, England)*, 34(15):2530–37.

Mandric, I. and Zelikovsky, A. (2015). ScaffMatch: Scaffolding algorithm based on maximum weight matching. *Bioinformatics (Oxford, England)*, 31(16):2632–38.

Marçais, G., Yorke, J. A., and Zimin, A. (2015). QuorUM: An Error Corrector for Illumina Reads. *PloS One*, 10(6):0130821.

Margulies, M., Egholm, M., Altman, W. E., Attiya, S., Bader, J. S., Bemben, L. A., and Berka, J. (2005). Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–80.

Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, 17(1):10–12.

Matchado, M. S., Lauber, M., Reitmeier, S., Kacprowski, T., Baumbach, J., Haller, D., and List, M. (2021). Network analysis methods for studying microbial communities: A mini review. *Computational and Structural Biotechnology Journal*, 19:2687–2698.

McCorrison, J. M., Venepally, P., Singh, I., Fouts, D. E., Lasken, R. S., and Methé, B. A. (2014). NeatFreq: Reference-free data reduction and coverage normalization for de novo sequence assembly. *BMC Bioinformatics*, 15(November):357.

McGreig, S. (2022). Angua3.

Meleshko, D., Hajirasouliha, I., and Korobeynikov, A. (2021). coronaSPAdes: From biosynthetic gene clusters to RNA viral assemblies. *Bioinformatics*, 38(1):1–8.

Mende, D. R., Waller, A. S., Sunagawa, S., Järvelin, A. I., Chan, M. M., Arumugam, M., Raes, J., and Bork, P. (2012). Assessment of metagenomic assembly using simulated next generation sequencing data. *PloS One*, 7(2):31386.

Menzel, P., Ng, K. L., and Krogh, A. (2016). Fast and sensitive taxonomic classification for metagenomics with kaiju. *Nature Communications*, 7(April):11257.

Mihara, T., Nishimura, Y., Shimizu, Y., Nishiyama, H., Yoshikawa, G., Uehara, H., Hingamp, P., Goto, S., and Ogata, H. (2016). Linking Virus Genomes with Host Taxonomy. *Viruses*, 8(3):66.

Miller, J. (2022). Mathematical Modeling Suggests Cooperation of Plant-Infecting Viruses - PMC. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9029262/.

Mims, C. (1981). Vertical Transmission of Viruses. *Microbiological Reviews*, 45(2):267–86.

Mirdita, M., Steinegger, M., Breitwieser, F., Söding, J., and Levy Karin, E. (2021). Fast and sensitive taxonomic assignment to metagenomic contigs. *Bioinformatics (Oxford, England)*, 37(18):3029–3031.

Mirdita, M., von den Driesch, L., Galiez, C., Martin, M. J., Söding, J., and Steinegger, M. (2017). Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Research*, 45(D1):D170–D176.

Mirebrahim, H., Close, T. J., and Lonardi, S. (2015). De Novo Meta-Assembly of Ultra-Deep Sequencing Data. *Bioinformatics (Oxford, England)*, 31(12):9–16.

Mistry, J., Chuguransky, S., Williams, L., Qureshi, M., Salazar, G. A., Sonnhammer, E. L. L., Tosatto, S. C. E., Paladin, L., Raj, S., Richardson, L. J., Finn, R. D., and Bateman, A. (2021). Pfam: The protein families database in 2021. *Nucleic Acids Research*, 49(D1):D412–D419.

Mizuno, C. M., Rodriguez-Valera, F., Kimes, N. E., and Ghai, R. (2013). Expanding the marine virosphere using metagenomics. *PLoS genetics*, 9(12):e1003987.

Modolo, L. and Lerat, E. (2015). UrQt: An efficient software for the unsupervised quality trimming of NGS data. *BMC Bioinformatics*, 16.

Moshiri, N. (2021). ViralMSA: Massively scalable reference-guided multiple sequence alignment of viral genomes. *Bioinformatics*, 37(5):714–716.

Muggli, M. D., Bowe, A., Noyes, N. R., Morley, P. S., Belk, K. E., Raymond, R., Gagie, T., Puglisi, S. J., and Boucher, C. (2017). Succinct Colored de Bruijn Graphs. *Bioinformatics (Oxford, England)*, 33(20):3181–87.

Nagarajan, N., Read, T. D., and Pop, M. (2008). Scaffolding and validation of bacterial genome assemblies using optical restriction maps. *Bioinformatics (Oxford, England)*, 24(10):1229–35.

Namiki, T., Hachiya, T., Tanaka, H., and Sakakibara, Y. (2012). MetaVelvet: An extension of velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Research*, 40(20):155.

National Library of Medicine (2023). BLAST Databases. https://www.nlm.nih.gov/ncbi/workshops/2023-08_BLAST_evol/databases.html.

National Library of Medicine (2024). NCBI Viral Assembly Datasets. https://www.ncbi.nlm.nih.gov/datasets/genome/?taxon=10239.

Nayfach, S., Páez-Espino, D., Call, L., Low, S. J., Sberro, H., Ivanova, N. N., Proal, A. D., Fischbach, M. A., Bhatt, A. S., Hugenholtz, P., and Kyrpides, N. C. (2021). Metagenomic compendium of 189,680 DNA viruses from the human gut microbiome. *Nature Microbiology*, 6(7):960–970.

Neri, U., Wolf, Y. I., Roux, S., Camargo, A. P., Lee, B., Kazlauskas, D., Chen, I. M., Ivanova, N., Zeigler Allen, L., Paez-Espino, D., Bryant, D. A., Bhaya, D., Krupovic, M., Dolja, V. V., Kyrpides, N. C., Koonin, E. V., Gophna, U., Narrowe, A. B., Probst, A. J., Sczyrba, A., Kohler, A., Séguin, A., Shade, A., Campbell, B. J., Lindahl, B. D., Reese, B. K., Roque, B. M., DeRito, C., Averill, C., Cullen, D., Beck, D. A., Walsh, D. A., Ward, D. M., Wu, D., Eloe-Fadrosh, E., Brodie, E. L., Young, E. B., Lilleskov, E. A., Castillo, F. J., Martin, F. M., LeCleir, G. R., Attwood, G. T., Cadillo-Quiroz, H., Simon, H. M., Hewson, I., Grigoriev, I. V., Tiedje, J. M., Jansson, J. K., Lee, J., VanderGheynst, J. S., Dangl, J., Bowman, J. S., Blanchard, J. L., Bowen, J. L., Xu, J., Banfield, J. F., Deming, J. W., Kostka, J. E., Gladden, J. M., Rapp, J. Z., Sharpe, J., McMahon, K. D., Treseder, K. K., Bidle, K. D., Wrighton, K. C., Thamatrakoln, K., Nusslein, K., Meredith, L. K., Ramirez, L., Buee, M., Huntemann, M., Kalyuzhnaya, M. G., Waldrop, M. P., Sullivan, M. B., Schrenk, M. O., Hess, M., Vega, M. A., O'Malley, M. A., Medina, M., Gilbert, N. E., Delherbe, N., Mason, O. U., Dijkstra, P., Chuckran, P. F., Baldrian, P., Constant, P., Stepanauskas, R., Daly, R. A., Lamendella, R., Gruninger, R. J., McKay, R. M., Hylander, S., Lebeis, S. L., Esser, S. P., Acinas, S. G., Wilhelm, S. S., Singer, S. W., Tringe, S. S., Woyke, T., Reddy, T., Bell, T. H., Mock, T., McAllister, T., Thiel, V., Denef, V. J., Liu, W.-T., Martens-Habbena, W., Allen Liu, X.-J., Cooper, Z. S., and Wang, Z. (2022). Expansion of the global RNA virome reveals diverse clades of bacteriophages. *Cell*, 185(21):4023–4037.e18.

Newcastle University IT Service (2023). IT Service (NUIT) - Newcastle University. https://services.ncl.ac.uk/itservice/research/hpc/.

NIASC and NIASC (2017). VirusMeta. NIASC.

Nijkamp, J., Winterbach, W., Broek, M., Daran, J.-M., Reinders, M., and Ridder, D. (2010). Integrating genome assemblies with MAIA. *Bioinformatics (Oxford, England)*, 26(18):433–39.

Nijkamp, J. F., Pop, M., Reinders, M. J., and Ridder, D. (2013). Exploring variation-aware contig graphs for (comparative) metagenomics using MaryGold. *Bioinformatics (Oxford, England)*, 29(22):2826–34.

Nooij, S., Schmitz, D., Vennema, H., Kroneman, A., and Koopmans, M. P. G. (2018). Overview of Virus Metagenomic Classification Methods and Their Biological Applications. *Frontiers in Microbiology*, 9.

Nothman, J. (2023). UpSetPlot documentation — upsetplot 0.8.0 documentation. https://upsetplot.readthedocs.io/en/stable/.

Nurk, S., Meleshko, D., Korobeynikov, A., and Pevzner, P. A. (2017). metaSPAdes: A New Versatile Metagenomic Assembler. *Genome Research*, 27(5):824–34.

O'Connell, J., Schulz-Trieglaff, O., Carlson, E., Hims, M. M., Gormley, N. A., and Cox, A. J. (2015). NxTrim: Optimized Trimming of Illumina Mate Pair Reads. *Bioinformatics (Oxford, England)*, 31(12):2035–37.

O'Leary, N. A., Wright, M. W., Brister, J. R., Ciufo, S., Haddad, D., McVeigh, R., Rajput, B., Robbertse, B., Smith-White, B., Ako-Adjei, D., Astashyn, A., Badretdin, A., Bao, Y., Blinkova, O., Brover, V., Chetvernin, V., Choi, J., Cox, E., Ermolaeva, O., Farrell, C. M., Goldfarb, T., Gupta, T., Haft, D., Hatcher, E., Hlavina, W., Joardar, V. S., Kodali, V. K., Li, W., Maglott, D., Masterson, P., McGarvey, K. M., Murphy, M. R., O'Neill, K., Pujar, S., Rangwala, S. H., Rausch, D., Riddick, L. D., Schoch, C., Shkeda, A., Storz, S. S., Sun, H., Thibaud-Nissen, F., Tolstoy, I., Tully, R. E., Vatsan, A. R., Wallin, C., Webb, D., Wu, W., Landrum, M. J., Kimchi, A., Tatusova, T., DiCuccio, M., Kitts, P., Murphy, T. D., and Pruitt, K. D. (2016). Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–745.

Ondov, B. D., Starrett, G. J., Sappington, A., Kostic, A., Koren, S., Buck, C. B., and Phillippy, A. M. (2019). Mash Screen: High-throughput sequence containment estimation for genome discovery. *Genome Biology*, 20(1):232.

Ondov, B. D., Treangen, T. J., Melsted, P., Mallonee, A. B., Bergman, N. H., Koren, S., and Phillippy, A. M. (2016). Mash: Fast genome and metagenome distance estimation using MinHash. *Genome Biology*, 17(1):132.

Ounit, R., Wanamaker, S., Close, T. J., and Lonardi, S. (2015). CLARK: Fast and accurate classification of metagenomic and genomic sequences using discriminative K-mers. *BMC Genomics*, 16.

Paez-Espino, D., Eloe-Fadrosh, E. A., Pavlopoulos, G. A., Thomas, A. D., Huntemann, M., Mikhailova, N., Rubin, E., Ivanova, N. N., and Kyrpides, N. C. (2016). Uncovering Earth's virome. *Nature*, 536(7617):425–430.

Pal, S. and Aluru, S. (2015). In search of perfect reads. *BMC Bioinformatics*, 16 Suppl 17.

Parras-Moltó, M., Rodríguez-Galet, A., Suárez-Rodríguez, P., and López-Bueno, A. (2018). Evaluation of bias induced by viral enrichment and random amplification protocols in metagenomic surveys of saliva DNA viruses. *Microbiome*, 6(1):119.

Paszkiewicz, K. and Studholme, D. J. (2010). De novo assembly of short sequence reads. *Briefings in Bioinformatics*, 11(5):457–72.

Paulino, D., Warren, R. L., Vandervalk, B. P., Raymond, A., Jackman, S. D., and Birol, I. (2015). Sealer: A scalable gap-closing application for finishing draft genomes. *BMC Bioinformatics*, 16.

Pecman, A., Adams, I., Gutiérrez-Aguirre, I., Fox, A., Boonham, N., Ravnikar, M., and Kutnjak, D. (2022). Systematic Comparison of Nanopore and Illumina Sequencing for the Detection of Plant Viruses and Viroids Using Total RNA Sequencing Approach. *Frontiers in Microbiology*, 13:883921.

Pecman, A., Kutnjak, D., Gutiérrez-Aguirre, I., Adams, I., Fox, A., Boonham, N., and Ravnikar, M. (2017). Next Generation Sequencing for Detection and Discovery of Plant Viruses and Viroids: Comparison of Two Approaches. *Frontiers in Microbiology*, 8:1998.

Peng, Q., Li, W., Zhou, X., Sun, C., Hou, Y., Hu, M., Fu, S., Zhang, J., Kundu, J. K., and Lei, L. (2023). Genetic Diversity Analysis of Brassica Yellows Virus Causing Aberrant Color Symptoms in Oilseed Rape. *Plants*, 12(5):1008.

Peng, Y. (2012). IDBA-UD: A de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics (Oxford, England)*, 28(11):1420–28.

Peng, Y., Dallas, M. M., Ascencio-Ibáñez, J. T., Hoyer, J. S., Legg, J., Hanley-Bowdoin, L., Grieve, B., and Yin, H. (2022). Early detection of plant virus infection using multispectral imaging and spatial–spectral machine learning. *Scientific Reports*, 12(1):3113.

Peng, Y., Leung, H. C., Yiu, S., and Chin, F. Y. (2011). Meta-IDBA: A de Novo Assembler for Metagenomic Data. *Bioinformatics (Oxford, England)*, 27(13):94–101.

Peng, Y., Leung, H. C., Yiu, S.-M., Lv, M.-J., Zhu, X.-G., and Chin, F. Y. (2013). IDBA-Tran: A more robust de novo de bruijn graph assembler for transcriptomes with uneven expression levels. *Bioinformatics (Oxford, England)*, 29(13):326–34.

Pérez-Rubio, P., Lottaz, C., and Engelmann, J. C. (2019). FastqPuri: High-performance preprocessing of RNA-Seq data. *BMC Bioinformatics*, 20(1):226.

Pericard, P., Dufresne, Y., Couderc, L., Blanquart, S., and Touzet, H. (2018). MATAM: Reconstruction of phylogenetic marker genes from short sequencing reads in metagenomes. *Bioinformatics (Oxford, England)*, 34(4):585–91.

Peyambari, M., Warner, S., Stoler, N., Rainer, D., and Roossinck, M. J. (2019). A 1,000-Year-Old RNA virus. *Journal of Virology*, 93(1).

Piro, V. C., Faoro, H., Weiss, V. A., Steffens, M. B., Pedrosa, F. O., Souza, E. M., and Raittz, R. T. (2014). FGAP: An automated gap closing tool.

Prjibelski, A., Antipov, D., Meleshko, D., Lapidus, A., and Korobeynikov, A. (2020). Using SPAdes De Novo Assembler. *Current Protocols in Bioinformatics*, 70(1):e102.

Qiagen (2023). RNeasy Kits. https://www.qiagen.com/us/products/discovery-and-translational-research/dna-rna-purification/rna-purification/total-rna/rneasy-kits.

Radford, A. D., Chapman, D., Dixon, L., Chantrey, J., Darby, A. C., and Hall, N. (2012). Application of next-Generation sequencing technologies in virology. *The Journal of General Virology*, 93(Pt 9):1853–68.

Rautiainen, M. and Marschall, T. (2020). GraphAligner: Rapid and versatile sequence-to-graph alignment. *Genome Biology*, 21(1):253.

Reddy, R. M., Mohammed, M. H., and Mande, S. S. (2014). MetaCAA: A Clustering-Aided Methodology for Efficient Assembly of Metagenomic Datasets. *Genomics*, 103(2-3):161–68.

Ren, J., Ahlgren, N. A., Lu, Y. Y., Fuhrman, J. A., and Sun, F. (2017). VirFinder: A novel k-mer based tool for identifying viral sequences from assembled metagenomic data. *Microbiome*, 5(1):69.

Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., Xie, X., Poplin, R., and Sun, F. (2020). Identifying viruses from metagenomic data using deep learning. *Quantitative Biology (Beijing, China)*, 8(1):64–77.

Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., Xie, X., and Sun, F. (2018). Identifying viruses from metagenomic data by deep learning.

Rfam team (2023). Index of /pub/databases/Rfam. https://ftp.ebi.ac.uk/pub/databases/Rfam/.

Robertson, G., Schein, J., Chiu, R., Corbett, R., Field, M., Jackman, S. D., and Mungall, K. (2010). De Novo Assembly and Analysis of RNA-Seq Data. *Nature Methods*, 7(11):909–12.

Rognes, T. (2011). Faster Smith-Waterman database searches with inter-sequence SIMD parallelisation. *BMC Bioinformatics*, 12(1):221.

Röhling, S., Linne, A., Schellhorn, J., Hosseini, M., Dencker, T., and Morgenstern, B. (2020). The number of k-mer matches between two DNA sequences as a function of k and applications to estimate phylogenetic distances. *PLOS ONE*, 15(2):e0228070.

Roldão, A., Silva, A., Mellado, M., Alves, P., and Carrondo, M. (2011). Viruses and Virus-Like Particles in Biotechnology. *Comprehensive Biotechnology*, pages 625–649.

Roossinck, M. J. (2010). Lifestyles of plant viruses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1548):1899–1905.

Roossinck, M. J. (2012). Plant virus metagenomics: Biodiversity and ecology. *Annual Review of Genetics*, 46(August):359–69.

Roossinck, M. J., Martin, D. P., and Roumagnac, P. (2015). Plant virus metagenomics: Advances in virus discovery. *Phytopathology*, 105(6):716–27.

Rosen, G., Garbarine, E., Caseiro, D., Polikar, R., and Sokhansanj, B. (2008). Metagenome Fragment Classification Using N-Mer Frequency Profiles. In *Advances in Bioinformatics 2008*.

Roux, S., Enault, F., Hurwitz, B. L., and Sullivan, M. B. (2015). VirSorter: Mining viral signal from microbial genomic data. *PeerJ*, 3.

Roux, S., Tournayre, J., Mahul, A., Debroas, D., and Enault, F. (2014). Metavir 2: New tools for viral metagenome comparison and assembled virome analysis. *BMC Bioinformatics*, 15.

Ruby, J., Bellare, P., and Derisi, J. L. (2013). PRICE: Software for the targeted assembly of components of (meta) genomic sequence data.

Sá, P. H., Miranda, F., Veras, A., Melo, D. M., Soares, S., Pinheiro, K., Guimarães, L., Azevedo, V., Silva, A., and Ramos, R. T. (2016). GapBlaster-A graphical gap filler for prokaryote genomes. *PloS One*, 11(5):0155327.

Sahlin, K., Vezzi, F., Nystedt, B., Lundeberg, J., and Arvestad, L. (2014). BESST–Efficient scaffolding of large fragmented assemblies. *BMC Bioinformatics*, 15(August):281.

Salmela, L., Mäkinen, V., Välimäki, N., Ylinen, J., and Ukkonen, E. (2011). Fast scaffolding with small independent mixed integer programs. *Bioinformatics (Oxford, England)*, 27(23):3259–65.

Salmela, L. and Rivals, E. (2014). LoRDEC: Accurate and efficient long read error correction. *Bioinformatics (Oxford, England)*, 30(24):3506–14.

Salmela, L. and Schröder, J. (2011). Correcting errors in short reads by multiple alignments. *Bioinformatics (Oxford, England)*, 27(11):1455–61.

Salmela, L., Walve, R., Rivals, E., and Ukkonen, E. (2017). Accurate self-correction of errors in long reads using de bruijn graphs. *Bioinformatics (Oxford, England)*, 33(6):799–806.

Sameith, K., Roscito, J. G., and Hiller, M. (2017). Iterative error correction of long sequencing reads maximizes accuracy and improves contig assembly. *Briefings in Bioinformatics*, 18(1):1–8.

Sanjuán, R., Nebot, M. R., Chirico, N., Mansky, L. M., and Belshaw, R. (2010). Viral Mutation Rates. *Journal of Virology*, 84(19):9733–9748.

Santiago-Rodriguez, T. M. and Hollister, E. B. (2022). Unraveling the viral dark matter through viral metagenomics. *Frontiers in Immunology*, 13:1005107.

Schneider, W. L., Sherman, D. J., Stone, A. L., Damsteegt, V. D., and Frederick, R. D. (2004). Specific detection and quantification of plum pox virus by real-time fluorescent reverse transcription-PCR. *Journal of Virological Methods*, 120(1):97–105.

Scholz, M., Lo, C.-C., and Chain, P. S. (2014). Improved assemblies using a source-agnostic pipeline for MetaGenomic assembly by merging (MeGAMerge) of contigs. *Scientific Reports*, 4(October):6480.

Schubert, M., Lindgreen, S., and Orlando, L. (2016). AdapterRemoval v2: Rapid adapter trimming, identification, and read merging. *BMC Research Notes*, 9(February):88.

Schulz, M. H., Weese, D., Holtgrewe, M., Dimitrova, V., Niu, S., Reinert, K., and Richard, H. (2014). Fiona: A parallel and automatic strategy for read error correction. *Bioinformatics (Oxford, England)*, 30(17):356–63.

Schulz, M. H., Zerbino, D. R., Vingron, M., and Birney, E. (2012). Oases: Robust de novo RNA-Seq assembly across the dynamic range of expression levels. *Bioinformatics (Oxford, England)*, 28(8):1086–92.

Segura, M. M., Kamen, A. A., and Garnier, A. (2011). Overview of current scalable methods for purification of viral vectors. *Methods in Molecular Biology*, 737:89–116.

Sellappan, L., Manoharan, S., Sanmugam, A., and Anh, N. T. (2022). 23 - Role of nanobiosensors and biosensors for plant virus detection. In Denizli, A., Nguyen, T. A., Rajendran, S., Yasin, G., and Nadda, A. K., editors, *Nanosensors for Smart Agriculture*, Micro and Nano Technologies, pages 493–506. Elsevier.

Sheikhizadeh, S. and Ridder, D. (2015). ACE: Accurate correction of errors using K-mer tries. *Bioinformatics (Oxford, England)*, 31(19):3216–18.

Shepard, S. S., Meno, S., Bahl, J., Wilson, M. M., Barnes, J., and Neuhaus, E. (2016). Viral deep sequencing needs an adaptive approach: IRMA, the iterative refinement meta-assembler. *BMC Genomics*, 17(September):708.

Shlemov, A. and Korobeynikov, A. (2019). PathRacer: Racing Profile HMM Paths on Assembly Graph. In Holmes, I., Martín-Vide, C., and Vega-Rodríguez, M. A., editors, *Algorithms for Computational Biology*, Lecture Notes in Computer Science, pages 80–94, Cham. Springer International Publishing.

Shrestha, R. K., Lubinsky, B., Bansode, V. B., Moinz, M. B., McCormack, G. P., and Travers, S. A. (2014). QTrim: A novel tool for the quality trimming of sequence reads generated using the roche/454 sequencing platform. In *BMC Bioinformatics 15*.

Simner, P. J., Miller, S., and Carroll, K. C. (2018). Understanding the Promises and Hurdles of Metagenomic Next-Generation Sequencing as a Diagnostic Tool for Infectious Diseases. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, 66(5):778–788.

Simpson, J. T., Wong, K., Jackman, S. D., Schein, J. E., Jones, S. J., and Birol, I. (2009). ABySS: A parallel assembler for short read sequence data. *Genome Research*, 19(6):1117–23.

Skewes-Cox, P., Sharpton, T. J., Pollard, K. S., and DeRisi, J. L. (2014). Profile hidden markov models for the detection of viruses within metagenomic sequence data. *PloS One*, 9(8):105067.

Sohn, J.-I. and Nam, J.-W. (2018). The present and future of de novo whole-genome assembly. *Briefings in Bioinformatics*, 19(1):23–40.

Sõmera, M., Fargette, D., Hébrard, E., Sarmiento, C., and ICTV Report Consortium (2021). ICTV Virus Taxonomy Profile: Solemoviridae 2021: This article is part of the ICTV Virus Taxonomy Profiles collection. *Journal of General Virology*, 102(12).

Song, L. and Florea, L. (2015). Rcorrector: Efficient and accurate error correction for illumina RNA-Seq reads. *GigaScience*, 4(October):48.

Soueidan, H., Maurier, F., Groppi, A., Sirand-Pugnet, P., Tardy, F., Citti, C., Dupuy, V., and Nikolski, M. (2013). Finishing bacterial genome assemblies with mix. *BMC Bioinformatics*, 14 Suppl 15.

Stahl, D. (1985). Characterization of a yellowstone hot spring microbial community by 5S rRNA sequences. *Applied and Environmental Microbiology*, 49(6):1379–84.

Stahl, D., Lane, D., Olsen, G., and Pace, N. (1984). Analysis of hydrothermal vent-associated symbionts by ribosomal RNA sequences. *Science (New York, N.Y.)*, 224(4647):409–11.

Steinegger, M., Meier, M., Mirdita, M., Voehringer, H., Haunsberger, S. J., and Soeding, J. (2019). HH-suite3 for fast remote homology detection and deep protein annotation.

Steinegger, M. and Söding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35(11):1026–28.

Steinegger, M. and Söding, J. (2018). Clustering huge protein sequence sets in linear time. *Nature Communications*, 9(1):2542.

Steinhauer, D. and Holland, J. (1987). Rapid evolution of RNA viruses. *Annual Review of Microbiology*, 41:409–33.

Sturm, M., Schroeder, C., and Bauer, P. (2016). SeqPurge: Highly-sensitive adapter trimming for paired-end NGS data. *BMC Bioinformatics*, 17.

Subudhi, S., Rapin, N., Dorville, N., Hill, J. E., Town, J., Willis, C. K. R., Bollinger, T. K., and Misra, V. (2018). Isolation, characterization and prevalence of a novel Gammaherpesvirus in Eptesicus fuscus, the North American big brown bat. *Virology*, 516:227–238.

Sukhorukov, G., Khalili, M., Gascuel, O., Candresse, T., Marais-Colombel, A., and Nikolski, M. (2022). VirHunter: A Deep Learning-Based Method for Detection of Novel RNA Viruses in Plant Sequencing Data. *Frontiers in Bioinformatics*, 2:867111.

Sun, K., Liu, Y., Zhou, X., Yin, C., Zhang, P., Yang, Q., Mao, L., Shentu, X., and Yu, X. (2022). Nanopore sequencing technology and its application in plant virus diagnostics. *Frontiers in Microbiology*, 13:939666.

Tampuu, A., Bzhalava, Z., Dillner, J., and Vicente, R. (2019). ViraMiner: Deep learning on raw DNA sequences for identifying viral genomes in human samples.

Tang, T., Liu, Y., Zhang, B., Su, B., and Li, J. (2019). Sketch distance-based clustering of chromosomes for large genome database compression. *BMC Genomics*, 20(S10):978.

Terriau, A., Albertini, J., Montassier, E., Poirier, A., and Le Bastard, Q. (2021). Estimating the impact of virus testing strategies on the COVID-19 case fatality rate using fixed-effects models. *Scientific Reports*, 11:21650.

The pandas development team (2020). Pandas-dev/pandas: Pandas. Zenodo.

Tobar-Tosse, F., Rodríguez, A. C., Vélez, P. E., Zambrano, M. M., and Moreno, P. A. (2013). Exploration of noncoding sequences in metagenomes. *PloS One*, 8(3):e59488.

Trifonov, V. and Rabadan, R. (2010). Frequency Analysis Techniques for Identification of Viral Genetic Data.

Turner, I., Garimella, K. V., Iqbal, Z., and McVean, G. (2018). Integrating long-range connectivity information into de bruijn graphs. *Bioinformatics (Oxford, England)*, 34(15):2556–65.

Uddin, M., Islam, M. K., Hassan, M. R., Jahan, F., and Baek, J. H. (2022). A fast and efficient algorithm for DNA sequence similarity identification. *Complex & Intelligent Systems*.

Van Rossum, G. and Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.

Varanda, C., Félix, M. d. R., Campos, M. D., and Materatski, P. (2021). An Overview of the Application of Viruses to Biotechnology. *Viruses*, 13(10):2073.

Vicedomini, R., Vezzi, F., Scalabrin, S., Arvestad, L., and Policriti, A. (2013). GAM-NGS: Genomic assemblies merger for next generation sequencing. *BMC Bioinformatics*, 14 Suppl 7.

Vieira, P., Subbotin, S. A., Alkharouf, N., Eisenback, J., and Nemchinov, L. G. (2022). Expanding the RNA virome of nematodes and other soil-inhabiting organisms. *Virus Evolution*, 8(1):veac019.

Vignuzzi, M., Stone, J. K., and Andino, R. (2005). Ribavirin and lethal mutagenesis of poliovirus: Molecular mechanisms, resistance and biological implications. *Virus Research*, 107(2):173–81.

Waite, D. W., Liefting, L., Delmiglio, C., Chernyavtseva, A., Ha, H. J., and Thompson, J. R. (2022). Development and Validation of a Bioinformatic Workflow for the Rapid Detection of Viruses in Biosecurity. *Viruses*, 14(10):2163.

Wan, Y., Renner, D. W., Albert, I., and Szpara, M. L. (2015). VirAmp: A galaxy-based viral genome assembly pipeline. *GigaScience*, 4.

Wang, I., Smith, D., and Young, R. (2000). Holins: The protein clocks of bacteriophage infections. *Annual Review of Microbiology*, 54:799–825.

Wang, Q., Fish, J. A., Gilman, M., Yanni Sun, C. B., Tiedje, J. M., and Cole, J. R. (2015). Xander: Employing a novel method for efficient gene-targeted metagenomic assembly. *Microbiome*, 3(August):32.

Wang, Q., Jia, P., and Zhao, Z. (2013). VirusFinder: Software for efficient and accurate detection of viruses and their integration sites in host genomes through next generation sequencing data. *PloS One*, 8(5):64465.

Wang, Y., Wang, K., Lu, Y. Y., and Sun, F. (2017). Improving Contig Binning of Metagenomic Data Using [Formula: See Text] Oligonucleotide Frequency Dissimilarity. *BMC Bioinformatics*, 18(1):425.

Wang, Z., Wang, Z., Lu, Y. Y., Sun, F., and Zhu, S. (2019). SolidBin: Improving metagenome binning with semi-supervised normalized cut. *Bioinformatics (Oxford, England)*.

Wanzeller, A. L. M., Souza, A. L. P., Azevedo, R. S. S., Júnior, E. C. S., Filho, L. C. F., Oliveira, R. S., Lemos, P. S., Júnior, J. V., and Vasconcelos, P. F. C. (2017). Complete Genome Sequence of the BeAn 58058 Virus Isolated from Oryzomys sp. Rodents in the Amazon Region of Brazil. *Genome Announcements*, 5(9):e01575–16.

Warnke-Sommer, J. and Ali, H. (2016). Graph mining for next generation sequencing: Leveraging the assembly graph for biological insights. *BMC Genomics*, 17.

Warren, R. (1980). Modified bases in bacteriophage DNAs. *Annual Review of Microbiology*, 34:137–58.

Waterhouse, P., Wang, M., and Lough, T. (2001). Gene silencing as an adaptive defence against viruses. *Nature*, 411(6839):834–42.

Wedemeyer, A., Kliemann, L., Srivastav, A., Schielke, C., Reusch, T. B., and Rosenstiel, P. (2017). An improved filtering algorithm for big read datasets and its application to single-cell assembly. *BMC Bioinformatics*, 18(1):324.

Wences, A. H. and Schatz, M. C. (2015). Metassembler: Merging and optimizing de novo genome assemblies. *Genome Biology*, 16.

Wetterstrand, KA. (2023). DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP). www.genome.gov/sequencingcostsdata.

Wheeler, T. J. and Eddy, S. R. (2013). Nhmmer: DNA homology search with profile HMMs. *Bioinformatics (Oxford, England)*, 29(19):2487–2489.

Wichels, A., Biel, S., Gelderblom, H., Brinkhoff, T., Muyzer, G., and Schütt, C. (1998). Bacteriophage diversity in the north sea. *Applied and Environmental Microbiology*, 64(11):4128–33.

Wick, R. R., Judd, L. M., Gorrie, C. L., and Holt, K. E. (2017). Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. *PLoS Computational Biology*, 13(6):1005595.

Wommack, K., Bhavsar, J., Polson, S. W., Chen, J., Dumas, M., Srinivasiah, S., Furman, M., Jamindar, S., and Nasko, D. J. (2012). VIROME: A standard operating procedure for analysis of viral metagenome sequences. *Standards in Genomic Sciences*, 6(3):427–39.

Wood, D. E. and Salzberg, S. L. (2014). Kraken: Ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, 15(3):46.

Wu, Y.-W., Rho, M., Doak, T. G., and Ye, Y. (2012). Stitching gene fragments with a network matching algorithm improves gene assembly for metagenomics. *Bioinformatics (Oxford, England)*, 28(18):363–69.

Wylie, K. M., Weinstock, G. M., and Storch, G. A. (2012). Emerging view of the human virome. *Translational Research: The Journal of Laboratory and Clinical Medicine*, 160(4):283–290.

Wylie, S. J., Tran, T. T., Nguyen, D. Q., Koh, S.-H., Chakraborty, A., Xu, W., Jones, M. G. K., and Li, H. (2019). A virome from ornamental flowers in an Australian rural town. *Archives of Virology*, 164(9):2255–2263.

Xie, Y., Wu, G., Tang, J., Luo, R., Patterson, J., Liu, S., and Huang, W. (2014). SOAPdenovo-Trans: De novo transcriptome assembly with short RNA-Seq reads. *Bioinformatics (Oxford, England)*, 30(12):1660–66.

Xu, G.-C., Xu, T.-J., Zhu, R., Zhang, Y., Li, S.-Q., Wang, H.-W., and Li, J.-T. (2019). LR_Gapcloser: A tiling path-based gap closer that uses long reads to complete genome assembly. *GigaScience*, 8(1).

Yang, S., Mao, Q., Wang, Y., He, J., Yang, J., Chen, X., Xiao, Y., He, Y., Zhao, M., Lu, J., Yang, Z., Dai, Z., Liu, Q., Yao, Y., Lu, X., Li, H., Zhou, R., Zeng, J., Li, W., Zhou, C., Wang, X., Shen, Q., Xu, H., Deng, X., Delwart, E., Shan, T., and Zhang, W. (2022). Expanding known viral diversity in plants: Virome of 161 species alongside an ancient canal. *Environmental Microbiome*, 17:58.

Yang, X., Charlebois, P., Gnerre, S., Coole, M. G., Lennon, N. J., Levin, J. Z., Qu, J., Ryan, E. M., Zody, M. C., and Henn, M. R. (2012). De novo assembly of highly diverse viral populations. *BMC Genomics*, 13(September):475.

Yang, Y. and Yooseph, S. (2013). SPA: A short peptide assembler for metagenomic data. *Nucleic Acids Research*, 41(8):91.

Yang, Y., Zhong, C., and Yooseph, S. (2015). SFA-SPA: A Suffix Array Based Short Peptide Assembler for Metagenomic Data. *Bioinformatics (Oxford, England)*, 31(11):1833–35.

Yao, G., Ye, L., Gao, H., Minx, P., Warren, W. C., and Weinstock, G. M. (2012). Graph accordance of next-Generation sequence assemblies. *Bioinformatics (Oxford, England)*, 28(1):13–16.

Ye, S. H., Siddle, K. J., Park, D. J., and Sabeti, P. C. (2019). Benchmarking Metagenomics Tools for Taxonomic Classification. *Cell*, 178(4):779–794.

Ye, Y. and Tang, H. (2009). An ORFome assembly approach to metagenomics sequences analysis. *Journal of Bioinformatics and Computational Biology*, 7(3):455–471.

Yeo, S., Coombe, L., Warren, R. L., Chu, J., and Birol, I. (2018). ARCS: Scaffolding genome drafts with linked reads. *Bioinformatics (Oxford, England)*, 34(5):725–31.

Yi, H. and Jin, L. (2013). Co-phylog: An assembly-free phylogenomic approach for closely related organisms. *Nucleic Acids Research*, 41(7):e75–e75.

Yoon, S., Kim, D., Kang, K., and Park, W. J. (2018). TraRECo: A greedy approach based de novo transcriptome assembler with read error correction using consensus matrix. *BMC Genomics*, 19(1):653.

Yu, G., Jiang, Y., Wang, J., Zhang, H., and Luo, H. (2018). BMC3C: Binning metagenomic contigs using codon usage, sequence composition and read coverage. *Bioinformatics (Oxford, England)*, 34(24):4172–79.

Zaharias, P. and Warnow, T. (2022). Recent progress on methods for estimating and updating large phylogenies. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 377(1861):20210244.

Zakrzewski, M., Bekel, T., Ander, C., Pühler, A., Rupp, O., Stoye, J., Schlüter, A., and Goesmann, A. (2013). MetaSAMS–a novel software platform for taxonomic classification, functional annotation and comparative analysis of metagenome datasets. *Journal of Biotechnology*, 167(2):156–65.

Zhao, G., Wu, G., Lim, E. S., Droit, L., Krishnamurthy, S., Barouch, D. H., Virgin, H. W., and Wang, D. (2017). VirusSeeker, a computational pipeline for virus discovery and virome composition analysis. *Virology*, 503(March):21–30.

Zhao, X. (2019). BinDash, software for fast genome distance estimation on a typical personal laptop. *Bioinformatics*, 35(4):671–673.

Zhao, Y., Tang, H., and Ye, Y. (2012). RAPSearch2: A fast and memory-efficient protein similarity search tool for next-Generation sequencing data. *Bioinformatics (Oxford, England)*, 28(1):125–26.

Zheng, W., Yang, L., Genco, R. J., Wactawski-Wende, J., Buck, M., and Sun, Y. (2019). SENSE: Siamese neural network for sequence embedding and alignment-free comparison. *Bioinformatics*, 35(11):1820–1828.

Zheng, Y., Gao, S., Padmanabhan, C., Li, R., Galvez, M., Gutierrez, D., Fuentes, S., Ling, K.-S., Kreuze, J., and Fei, Z. (2017). VirusDetect: An automated pipeline for efficient virus discovery using deep sequencing of small RNAs. *Virology*, 500(January):130–38.

Zhou, Q., Su, X., Wang, A., Xu, J., and Ning, K. (2013). QC-Chain: Fast and holistic quality control method for next-Generation sequencing data. *PloS One*, 8(4):60234.

Zhu, B.-H., Song, Y.-N., Xue, W., Xu, G.-C., Xiao, J., Sun, M.-Y., Sun, X.-W., and Li, J.-T. (2016). PEP_scaffolder: Using (homologous) Proteins to Scaffold Genomes. *Bioinformatics (Oxford, England)*, 32(20):3193–95.

Zielezinski, A., Girgis, H. Z., Bernard, G., Leimeister, C.-A., Tang, K., Dencker, T., Lau, A. K., Röhling, S., Choi, J. J., Waterman, M. S., Comin, M., Kim, S.-H., Vinga, S., Almeida, J. S., Chan, C. X., James, B. T., Sun, F., Morgenstern, B., and Karlowski, W. M. (2019). Benchmarking of alignment-free sequence comparison methods. *Genome Biology*, 20(1):144.

Zimin, A. V., Smith, D. R., Sutton, G., and Yorke, J. A. (2008). Assembly Reconciliation. *Bioinformatics (Oxford, England)*, 24(1):42–45.