

ANALYZING MARKER GENE DIVERSITY USING AN AUTOMATED
PHYLOGENETIC TOOL: AUTOPHY

by

Deepika Prasad

A thesis submitted to the Faculty of the University of Delaware in partial
fulfillment of the requirements for the degree of Master of Science in Bioinformatics
and Computational Biology

Spring 2017

© 2017 Deepika Prasad
All Rights Reserved

ANALYZING MARKER GENE DIVERSITY USING AN AUTOMATED
PHYLOGENETIC TOOL: AUTOPHY

by

Deepika Prasad

Approved: _____
Shawn Polson, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: _____
Kathleen F. McCoy, Ph.D.
Chair of the Department of Computer and Information Sciences

Approved: _____
Babatunde A. Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____
Ann L. Ardis, Ph.D.
Senior Vice Provost for Graduate and Professional Education

ACKNOWLEDGMENTS

I want to express my sincerest gratitude to Professor Shawn Polson for his patience, continuous support, and enthusiasm. I could not have imagined having a better mentor to guide me through the process of my Master's thesis.

I would like to thank my committee members, Professor Eric Wommack and Honzhan Huang for their insightful comments and guidance in the thesis.

I would also like to express my gratitude to Barbra Ferrell for asking important questions throughout my thesis research and helping me with the writing process.

I would like to thank my friend Sagar Doshi, and lab mates Daniel Nasko, and Prasanna Joglekar, for helping me out with data, presentations, and giving important advice, whenever necessary.

Last but not the least, I would like to thank my family, for being my pillars of strength.

TABLE OF CONTENTS

LIST OF TABLES	v
LIST OF FIGURES	vii
ABSTRACT	x
Chapter	
1 INTRODUCTION	1
1.1. Viruses	1
1.2. Markers and phylogenetics	2
1.3. Marker genes of special interest	3
1.4. Introduction to VIROME	8
1.5. Need for tools to analyze markers	10
2 DEVELOPMENT OF AUTOPHY	13
3 VALIDATION FOR AUTOPHY	31
3.1 Part – 1: Validation of codes in the pipeline	31
3.2 Part – 2: Exploring AutoPhy results	55
3.2.1 Background	56
3.2.2 Methodology exploration	58
3.2.3 Results.... ..	59
3.2.4 Discussion	64
4 CONCLUSION	69
REFERENCES	73

LIST OF TABLES

Table 2.1: lists the steps and names given within the wrapper for each of them. The names of the steps will be repeatedly used in various parts of the thesis. The steps listed in this table are the same step names given in the wrapper script.....	15
Table 3.1: Input and output files for AP – 1.....	34
Table 3.2: Input and output files for AP – 2.....	35
Table 3.3: Input and output files for AP – 3.....	37
Table 3.4: Input and output files for AP – 4.....	38
Table 3.5: Input and output files for AP – 5.....	40
Table 3.6: Input and output files for AP – 6.....	41
Table 3.7: Input and output files for AP – 7; output shown for different trimmed regions.....	44
Table 3.8: Input and output files for AP – 8.....	46
Table 3.9: Input and output files for AP – 9.....	48
Table 3.10: Range of threshold tests.....	49
Table 3.11: Input and output files for AP – 10.....	51
Table 3.12: Input and output files for AP – 13.....	52
Table 3.13: logic of AP–L–1.....	53
Table 3.14: Input and Output of AP – 12.....	53
Table 3.15: Table listing the tools used in both protocols.....	58
Table 3.16: AutoPhy statistics.....	59
Table 3.17: Manually curated tree statistics.....	59

Table 3.18: Snapshot alignment statistics.....	68
--	----

LIST OF FIGURES

Figure 1.1: Phylogenetic analysis on the alignment of sequences showed that clusters of sequences with the same residue at position 762 claded together (Source: Schmidt et al., 2014; adapted from Rachel Marine).....	6
Figure 1.2: Flowchart of the virome pipeline. After sequence analysis conducted using BLASTP, If the sequence is a significant UniRef hit with a significantly high e-value then it is put in the “known bin”, and if the homologue has a meaningful annotation then it is considered a functional protein. Sequences with hits only with MgOl are placed in the Environmental protein bin and then classified into “only microbial” or “only viral” hits. If the peptide has no hit against MgOl or UniRef100, it is considered an ORFan (novel gene). (Source: Wommack and Polson et al., 2012).....	9
Figure 2.1: Overall conceptual flowchart of AutoPhy. Pipeline starts with input data and reformatting of the headers; after the reformatting, the sequences go through multiple sequence alignment; alignment then goes into the trim suite; phylogenetic analysis is performed on the conditioned data, through which a newick file is produced. The original headers are then replaced into the substituted names in the newick file.....	16
Figure 2.2: Describes the logic of selection of paths within the pipeline by the decision making script; helps AutoPhy take clustering or non-clustering path depending on size of dataset.....	17
Figure 2.3: logical flow of data through of reformatting headers and combining of reference sequence file; AP – 1, AP – 2, AP-5.....	19
Figure 2.4: flowchart explaining the change in reference headers; AP – 3, AP – 4, AP – 5.....	20
Figure 2.5: multiple sequence alignment in phylip format.....	22
Figure 2.6: The flowchart shows the different processes that are a part of the Trim suite; 1. Trimming script - Chop the sequence to the required region keeping the reference sequence as a basis; 2. script computes the ratio of	

the aligned region with respect to the cut off (Start and End coordinates) given; 3. Select the sequences – with a ratio above the threshold level.....	23
Figure 2.7: Describing logic of 1. trimming script flowing into, 2. calculates the threshold, moving onto 3. selects the sequences based on the threshold recorded for each sequence and 4. Converts everything back to the phylip format to be pushed further into the pipeline.....	24
Figure 2.8: An example of a trimmed sequence; EG = external gaps.....	25
Figure 2.9: flowchart showing the flow of data in clustering path.....	27
Figure 2.10: conceptual flowchart of the data from the trim suite to the rest of the pipeline.....	30
Figure 3.1: cutoff parameter applied within logic code.....	32
Figure 3.2: cutoff parameter applied within logic code.....	33
Figure 3.3: cutoff parameter not applied not utilized as the number of sequences was below the cutoff variable.....	33
Figure 3.4: cutoff parameter not applied as the number of sequences was below the cutoff variable.....	34
Figure 3.5: Snapshot of a clade of a consisting of leucines within the tree with singletons.....	56
Figure 3.6: cladogram of result without singletons.....	60
Figure 3.7: phylogram of AutoPhy result (with 725 sequences) including reference sequences.....	61
Figure 3.8: phylogram of phylogenetic tree without singletons (with 300 sequences) including reference sequences.....	62
Figure 3.9: cladogram of phylogenetic tree without singletons.....	63
Figure 3.10: This figure shows the three trees of importance in the study and a comparison. The cladogram on the extreme left shows the distribution of sequences according to the amino acid at position 762; the figure on the extreme right shows the distribution of sequences in the phylogenetic tree in the based on the amino acid and source of the sequence.....	65

Figure 3.11: snapshot of alignment in Geneious, blue column shows amino acids in the position 762 in different sequences.....67

ABSTRACT

Marker genes can be used to find biological insights from metagenomic data. Due to their expected conservation through evolution it is possible to identify them within a population. The most significant example is that of 16S rRNA marker gene which is widely used amongst the scientific community for bacteria. A universal marker gene for viral populations does not exist, as there is no one gene that is present on all viruses. There are a number of group specific marker genes for viruses, some of them essential for important functions such as replication. These marker genes within viral metagenomes can be analyzed to give insight into the biology of a viral assemblage; these tools are needed to automate the analysis of such data, which can help analyze the evolutionary relationship amongst viruses. In this project we develop AutoPhy, a tool developed to analyze marker gene diversity through the automated development of a phylogenetic tree from metagenomics data. This analytical tool integrates other tools and perl scripts on the shell environment, to streamline sequences chosen as marker genes of interest in metagenomes. It helps produce a tree in the newick format of the marker gene of interest from the sequences submitted. A unique suite of scripts makes up the pipeline that is involved in finding this region of interest using reference sequences, and specifically choosing the best sequences for the purpose. As a part of the study, the various scripts used in the pipeline including the trim suite were validated using mock data to check for their accuracy. In addition, the AutoPhy output was scrutinized using a manually created gold standard DNA

Polymerase A tree. The tool was produces a phylogenetic tree with expected structure, but there remains potential for improvements. The output generated requires fine-tuning in terms of identification and removal of artifactual deep branches and addition of an iterative step to better align and group some sequences. With future improvements, the tool will be implemented as a part of the VIROME pipeline.

Chapter 1

INTRODUCTION

1.1. Viruses

Viruses are an important part of the environment and the extent of their diversity and the roles they play, are still largely unknown. After years of research, enough evidence was collected to suggest that viruses were not just tiny pieces of life but a more significant part of the ecosystem. Viruses infect all domains of life and exist wherever cellular life can be found [1].

Marine virology has become an important topic of study, with concentrations of viruses at approximately 10^7 per milliliters of surface seawater [2]. The majority of these are bacteriophage (phage), viruses that infect bacteria. Phages are known to control bacterial population and bring considerable changes in their communities. Horizontal gene transfer can impact microbial processes like resistance and metabolism [2].

Bacteriophages mediate transduction, which is a mechanism of genetic exchange. Horizontal gene transfer is predominantly responsible for the evolution of antibiotic resistance bacteria through conjugation, transformation and transduction. Specialized transduction takes place only in temperate phages, whereas both lytic and temperate cells can undergo generalized transduction. Twenty-seven sequenced metagenomes were explored and it was observed that there were a high proportion of mobile genetic elements in addition to phages, among different microbial communities. When the proportion of these was so high, it was suggested that this might contribute to horizontal gene transfer [18].

Viruses can be discovered using a variety of techniques. The conventional methods include electron microscopy, cell culture, serology and inoculation studies. These techniques each have their own restrictions. Advances in technology have led to techniques such as microarray, hybridization-based methods and PCR amplification and shotgun metagenomes [3]. Shotgun metagenomics is able to significantly better explain structure and function of the naturally occurring communities, in comparison to the culture dependent methods [18]. A technological advance in the form of metagenomics has resulted in a more amalgamated study of microbial populations, by analyzing the nucleotide sequence in an entire sample [3]. Studies have shown that metagenomics was also able to avoid many of the limitations of the conventional methods. This can be illustrated with the “great plate-count anomaly”. It was observed that the diversity observed under the microscope, was not captured in cultures in petri plates or test tubes. All microbes could not grow on media; one of the reasons being microbes grow in a community, continuously exchanging genes, chemicals, and metabolic products. Metagenomics on the other hand, is able to in a sense keep the community together, and investigate the community at their source. They are sampled collectively which in turn helps keep the complexity and dynamics of their interactions intact [20]. This method has been able to describe the diversification of the environment, in a more holistic sense [4].

1.2. Markers and phylogenetics

In bacterial and other cellular life, quantification and getting a grasp on the vastness of the community has been achieved through the discovery of a universal marker, the SSU rRNA gene. When it was suggested that gene sequences could

possibly be used as time lines to elucidate phylogenetic relationships, the concept of rRNA genes helped with the purpose and was used to explain the three domains of life. The possibility of the ability to store rRNA sequences in the database and being able to reconstruct phylogeny using them was an important find in this world [21].

Using simplified marker gene studies is a more economical method of assessing the diversity of sequences [5] but in viruses, one of the leading problems is the lack of a universal marker gene. Higher probabilities of gene transfer events lead to difficulty in assessing the diversity [6]. Due to their polyphyletic origins, looking for marker genes group-specific level can possibly help with better determination of potentiality of the marker [7]. Marker genes of viral diversity must have the following qualities: ability to be widely distributed and also within the population being investigated, importance in viral biology (for e.g., essential for replication), ability in assisting the analysis of diversity and mapping genetic characteristics, presence in reference databases [8].

One of the most convenient ways to analyze markers is looking at phylogenetic relationships. Phylogenetic trees and their relationship defining branches are an important part of numerous areas of biology. The phylogenetic frame of reference has been able to conclude many patterns of evolution, including nitrogen fixing symbioses, mustard oil production, etc. Phylogenetic reconstruction can be applied to find history of evolution of genes [19]. It is always important to perform extensive phylogenetic analyses to the data [9].

1.3. Marker genes of special interest

Marker genes are good tools that can be used in a variety of research areas. They can especially be used extensively in analyzing metagenomic data. Marker genes

tend to be more phylogenetically conserved and the likeliness of them having undergone horizontal gene transfer is also supposed to be low. If specifically conserved genes are chosen, their phylogenetic signal can be exploited [10].

There are a number of markers/signature genes available for various virus groups. The marker gene product of cyanophages belonging to the *Myoviridae* family is g20 (major capsid protein) [23] is another marker for T4 members of freshwater cyanophages, *Phycodnaviridae*, and *Gokushovirinae* subfamily, of the *Myoviridae* family [5]; tail sheath protein, g91, one of the many markers for cyanophages of the *Myoviridae* family, that infect *Microcystis aeruginosa*[24]. The various photosynthetic proteins (psbA, psbD) are also seen as indicators of cyanophages [25]. Phosphate starvation protein derived from phoH has also been seen as a potential marker for cyanophages. The g43 signature gene, which gives the product DNA polymerase, is a possible marker gene for the T4-like members of *Myoviridae*. PolB gene, which codes for the DNA polymerase protein is seen as a possible marker for *Phycodnaviridae*. Gene RdRp gives the product RNA-dependent RNA polymerase seen in RNA viruses. Syn9_g101 coding for the gene product putative tail fiber is a possible occupant of Cyanophage and it belongs to the *Myoviridae* family. CobS codes for putative porphyrin biosynthetic protein and it is also a marker targeting the cyanophages of the *Myoviridae* [5]. T7mcp coding for major capsid protein targets the T7-like members of *Podoviridae*. PolA gene or DNA polymerase has been seen as a potential indicator for *Podoviridae* [11].

Schmidt et al., 2014 extensively analyzed DNA polymerases family A genes, which are found in tailed bacteriophages. DNA polymerase is an important component in DNA replication and can be found commonly across environments [11].

In most metagenomic sequences of virioplankton it was seen that three of the motifs were conserved in the DNA pol gene; Asparagine (binds catalytic magnesium), Glutamine (stabilizes enzyme and stops integration of ribonucleotides), Arginine at position 705, 710, and 712 respectively, are conserved residues in motif A. Arginine, Lysine, Phenylalanine and Tyrosine at position 754, 758, 762, and 766 respectively are conserved in motif B. It was seen that all of these residues and motif C (positions 881-883) were highly conserved, with an exception of the amino acid at position 762. The residue number 762 was chosen on the basis of reference sequence *E.coli* Pol I gene product, and it was observed that phenylalanine occurred in the wild type. Leucine mutation at this position was seen to be more common in comparison to tyrosine, amongst the virioplankton.

When phylogenetic analysis was performed on the alignment to observe the clustering of the sequences within the tree, it was seen that sequences with the same amino acid mutation or same amino acid at position 762 were clustering together. The analysis helped to observe the evolutionary distance amongst the different clades and the diversity of each one of them (Figure 1.1).

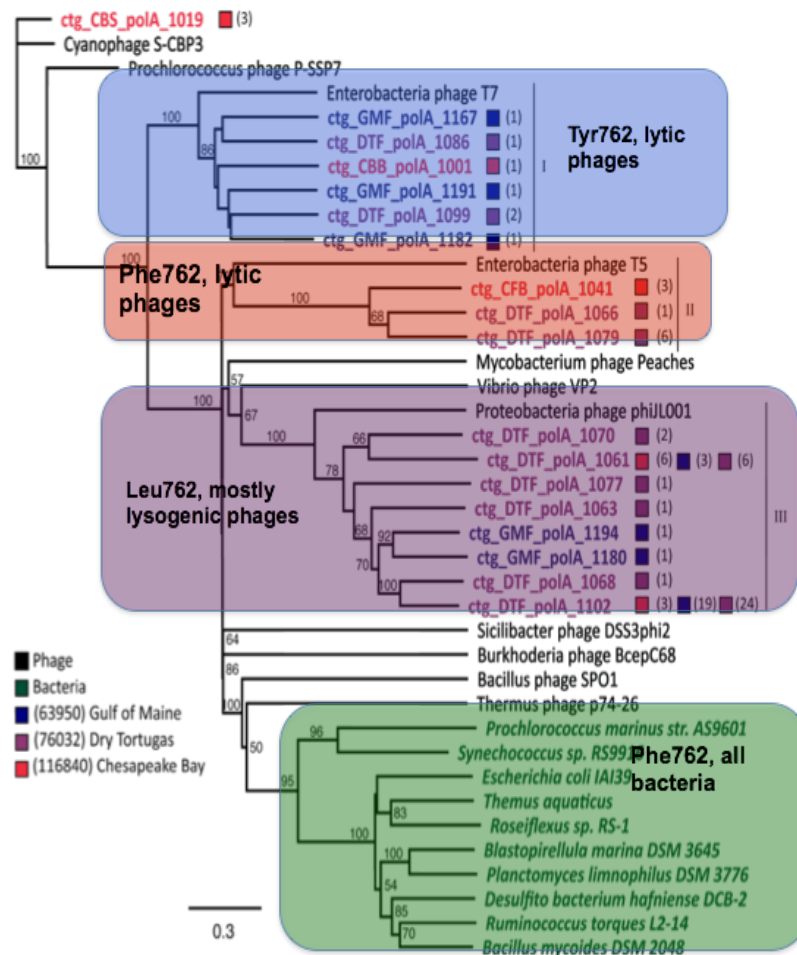


Figure 1.1: phylogenetic analysis on the alignment of sequences showed that clusters of sequences with the same residue at position 762 claded together (Source: Schmidt et al., 2014; adapted from Rachel Marine)

Lysogenic phages seemed to contain the leucine substitution at position 762, and lytic phages primarily had the tyrosine substitution and wild type phenylalanine, indicating that this position could be a potential indicator of the lifestyle followed by the phage [11].

Another paper of importance that involved an important marker gene find, was Sakowski et al., 2013 who discovered another potential marker of importance which were the Ribonucleotide reductases (RNRs). These enzymes are divided into 3 classes, based on their Oxygen dependence. It was seen that RNR contained a number of qualities, which made it an attractive marker gene for the study. They were also present in dsDNA viruses, which was an added advantage. It captured the diversity of viroplankton in way that was not possible with other genes. The main importance of this enzyme was that it was involved in DNA replication, because of which the potential of RNR as a marker gene was considered high. In addition, it was able to support “kill the winner” hypothesis which basically indicated that podoviruses which infect a specific spectrum of viruses, was found abundantly found in bacterioplankton which was dominant within the population of viroplankton [8]. “Kill the winner” hypothesis, in this context describes the effect of accessibility of resources, influence of predators and viruses, on viroplankton [21].

The information obtained from these marker genes are going to be used as reference sequences for a tool which will be explained in later sections, and this tool is potentially to be integrated as a middle layer in VIROME.

1.4. Introduction to VIROME

Viral Informatics Resource for Metagenome Exploration or VIROME (<http://virome.dbi.udel.edu/>) is a tool (Figure 1.2) developed to analyze and explore metagenomes using a web-based interface [13]. It consists of applications that are optimized to improve the analysis of viral metagenomes. The pipeline performs quality screening, sequence analysis, and describes sequence homology. The application interface helps users explore their results and bin sequences for further analysis [13].

In the sequence quality steps, to summarize, duplicate sequences are removed, any RNA homologs and contaminants that are present are also removed from the sequence libraries. After sequence quality steps, the sequences are analyzed and annotated. As shown in Figure 1.2, UniRef 100 knowledgebase [28] is used to obtain metagenomic sequences with similarity to known databases using BLASTp. The environmental sequences are detected using MgOl database, which consists of microbial and viral sequences, and which is then divided into microbial hits and viral hits.

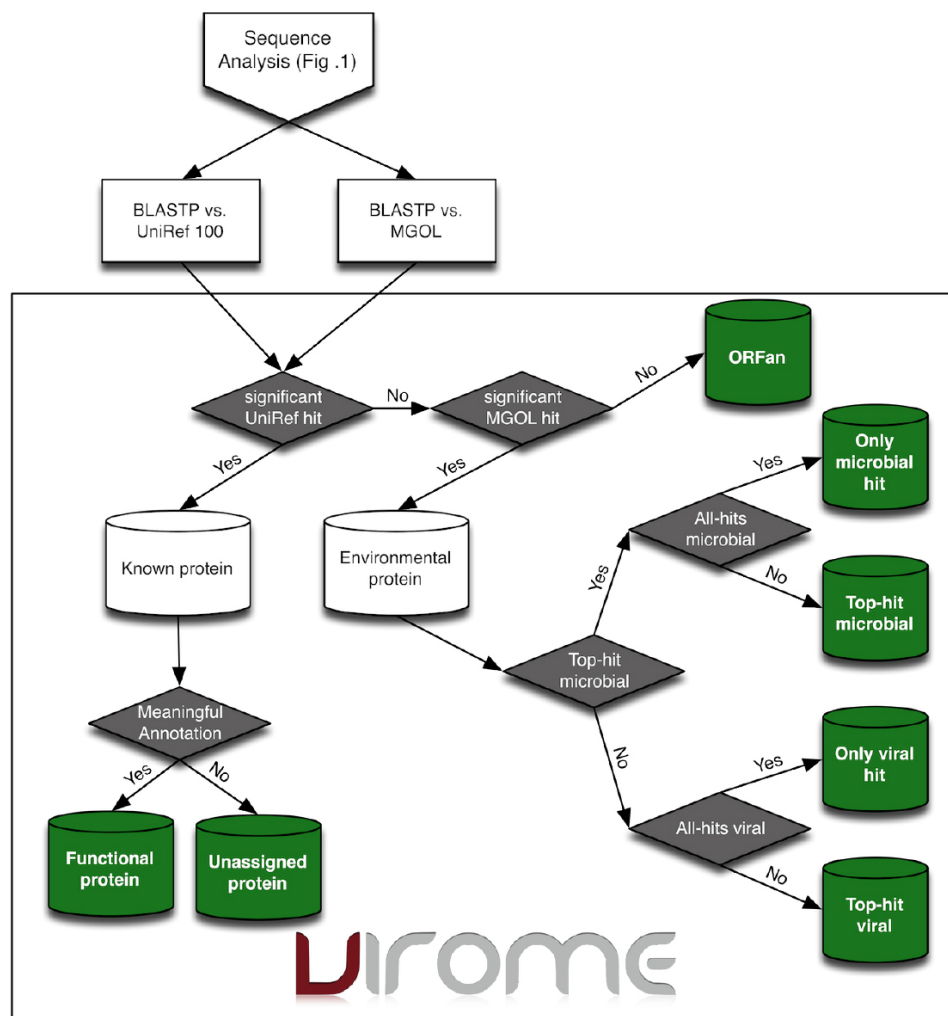


Figure 1.2: Flowchart of the virome pipeline. After sequence analysis conducted using BLASTP, If the sequence is a significant UniRef hit with a significantly high e-value then it is put in the “known bin”, and if the homologue has a meaningful annotation then it is considered a functional protein. Sequences with hits only with MgOl are placed in the Environmental protein bin and then classified into “only microbial” or “only viral” hits. If the peptide has no hit against MgOl or UniRef 100, it is considered an ORFan (novel gene). (Source: Wommack and Polson et al., 2012)

Metagenomes Online is a database that contains 270 metagenome libraries (as of April 6th, 2017). They can be used to find homologous sequences within the environment and this could in turn, help find novel sequences. There are also sequences of eukaryotic, viral and microbial origin. All the data in the database was predicted using shotgun metagenome sequences.

1.5. Need for tools to analyze markers

There is a need to build tools that will be able to find markers in datasets using reference sequences, automatically, in addition to the regions of interest. Schmidt et al (2014) observed that the region of interest (around position 762) had to be specifically extracted for analysis. This specific extraction helps in streamlining the marker selection process, finding conserved residues (if present), and mutations of interest. Finding evolutionary relationships, using reference sequences while taking advantage of trimmed regions helps in the overall process of building a clean tree.

An attempt is being made to provide a more automatic means of recreating the insights found through manual methods by Schmidt et al (2014), and Sakowski et al (2013). In this thesis, AutoPhy is described as an analytical tool that was designed to be a potential marker finder, using reference sequences. AutoPhy was started with the idea of an automated workflow, which would identify, process, and analyze marker genes in shotgun metagenome data on VIROME, to provide phylogenetic trees in the form of a newick format file.

METAVIR [22] is the only other tool available for automated phylogenetic tree generation, using reference sequences. A variety of marker genes are available against which trees can be built and can be requested for, using the website.

This tool uses the available marker gene information for viruses and develops phylogenetic trees. The tool has a number of projects available, each of which has a studied group listed in addition to a list of reference sequences. The source of the reference sequences is Pfam (link to the marker on Pfam [26], not available on website currently). As per the information available, it is not known when the reference sequences were updated last and on what criteria they were chosen to be able to represent the specific marker gene. It is not mentioned, if the sequences displayed on the phylogenetic tree, are selected based on any criteria or if complete sequences have been used in the tree.

AutoPhy is meant to actively examine data provided and use different sets of reference sequences, depending on the marker analyzed in the dataset. The tool meant to be an extension to the VIROME platform and before the sequences are submitted to the tool, the sequences are meant to undergo a few strict upstream processes.

After the ORFs have been annotated using UniRef 100 in VIROME pipeline, each ORF will have information attached to it, such as gene description and function and recorded in the VIROME database. The database is updated with the release of a new library. For AutoPhy depending on the markers being investigated, candidate ORFs matching a marker gene of interest is extracted by searching the marker gene keyword against the database. The sequences would have to be corroborated with the function that they have been assigned. If the example of DNA polymerase were taken, the sequences would be verified of their function using a script to confirm that a sequence is actually the sequence or protein of interest that is, it is actually a DNA polymerase. In the case of a DNA polymerase the prevalence of

the sequence can be confirmed with looking at the 762 (dnasko/dna_pola_762_caller.git) position that has been discussed in the beginning of Introduction. If the sequence does not have the necessary amino acid at position of interest, then the sequence would be removed from being analyzed further downstream. These highly validated sequences will then be forwarded to AutoPhy for the development of a phylogenetic tree. The tool AutoPhy will be discussed in further detail in the next chapter: “Chapter – 2: Development of AutoPhy”.

Chapter 2

DEVELOPMENT OF AUTOPHY

Phylogenetic trees can be remarkably good indicators of evolutionary relationships among different species. AutoPhy was inspired by the idea of automating the development of phylogenetic trees from a shotgun metagenome. AutoPhy uses this idea to take the task of finding markers, forward. The hope is to be able to confirm the usability and importance of regions of interest within sequences in the dataset and quantify them.

When sequences are being submitted to different tools to develop a phylogenetic tree, various factors are not taken into proper consideration, such as, number of sequences, length of sequences, importance of their alignment, specific regions of interest, etc. AutoPhy has been developed to account for all the above factors.

There are different parts to the tool or pipeline, which will be explained in detail in the following chapter.

Three tools and a number of scripts have been used in the pipeline to optimize results and affirm the plausibility of the marker gene. The tool has a number of adjustable parameters to help with this purpose. The data submitted to the pipeline is mainly metagenomic data that will be annotated by VIROME. The scripts have been written in Perl and the wrapper script was used to run the all the scripts consecutively. Each script produces intermediate output files, which can be used to analyze different

parts of the result. The ultimate outcome of this pipeline is to analyze the dataset to build a phylogenetic tree delivered as a newick file

AutoPhy is a tool with data conditioning steps and other tools, which help prepare the data for various steps within the pipeline, improving the eventual quality of the resultant phylogenetic tree.

Table 2.1 in the next page, summarises the role and position of each of the steps within the pipeline.

Table 2.1: lists the steps and names given within the wrapper for each of them. The names of the steps will be repeatedly used in various parts of the thesis. The steps listed in this table are the same step names given in the wrapper script

Name of Step	Script name	Role of script
AP-L-0	autophy-log-1_seqs_above_cutoff.pl	Decision making
AP-1	autophy-01-query_sub_tab.pl	Scripts to reformat headers for smooth flow of data
AP-2	autophy-02-query_head_sub.pl	
AP-3	autophy-03-ref_sub_tab.pl	
AP-4	autophy-04-ref_head_sub.pl	
AP-5	autophy-03-1-ref_sub_tab.pl	
AP-6	autophy-05-combine_ref_que.pl	Combining data
AP-T-1	Muscle implementation	Multiple sequence alignment (MSA)
AP-7	autophy-06-trim_reg_int.pl	Trim suite (trimming and conditioning of sequences)
AP-8	autophy-07-ratio_calc_for_selseqs.pl	
AP-9	autophy-08-sel_seqs.pl	
AP-T-2	USEARCH implementation	Clustering
AP-10	autophy-choose_REF_seqs.pl	Restructuring of the organization of sequences within query file for future MSA
AP-11	autophy-13_combine_ref_que.pl	
AP-L-1	autophy-log-2_sel_seqs_less.pl	Decision making
AP-13	autophy-09-format_fa-phyl.pl	Fasta to phylip conversion
AP-T-3	RAxML implementation	Phylogenetic analysis
AP-12	autophy-10-header_returner.pl	Reformatting of headers

Figure 2.1 shows the important steps of the pipeline.

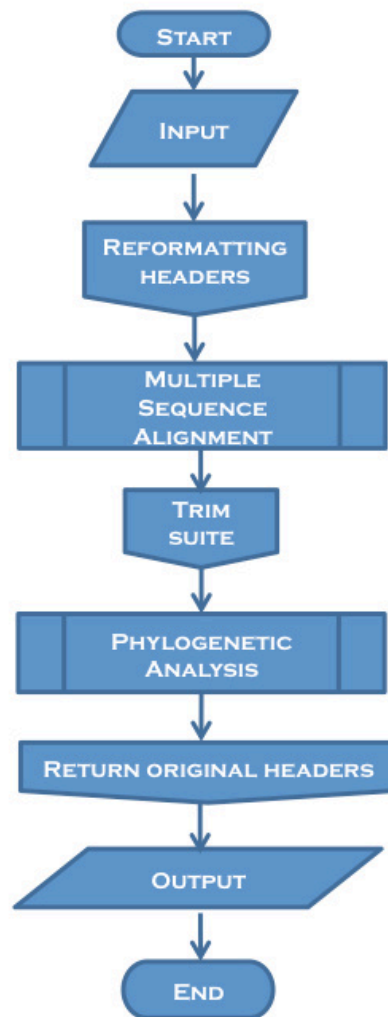


Figure 2.1: Overall conceptual flowchart of AutoPhy. Pipeline starts with input data and reformatting of the headers; after the reformatting, the sequences go through multiple sequence alignment; alignment then goes into the trim suite; phylogenetic analysis is performed on the conditioned data, through which a newick file is produced. The original headers are then replaced into the substituted names in the newick file.

AutoPhy, the pipeline (Figure 2.1) for building phylogenetic trees begins with sequences submitted to the pipeline in FASTA format. The path taken from here depends on a specific condition: the number of sequences submitted as a part of the dataset.

AP-L-0 (autophy-log-l_seqs_above_cutoff.pl): The decision-making script (Figure 2.2) counts the number of lines within the query file starting with the carat sign, and directs the numbers to another file.

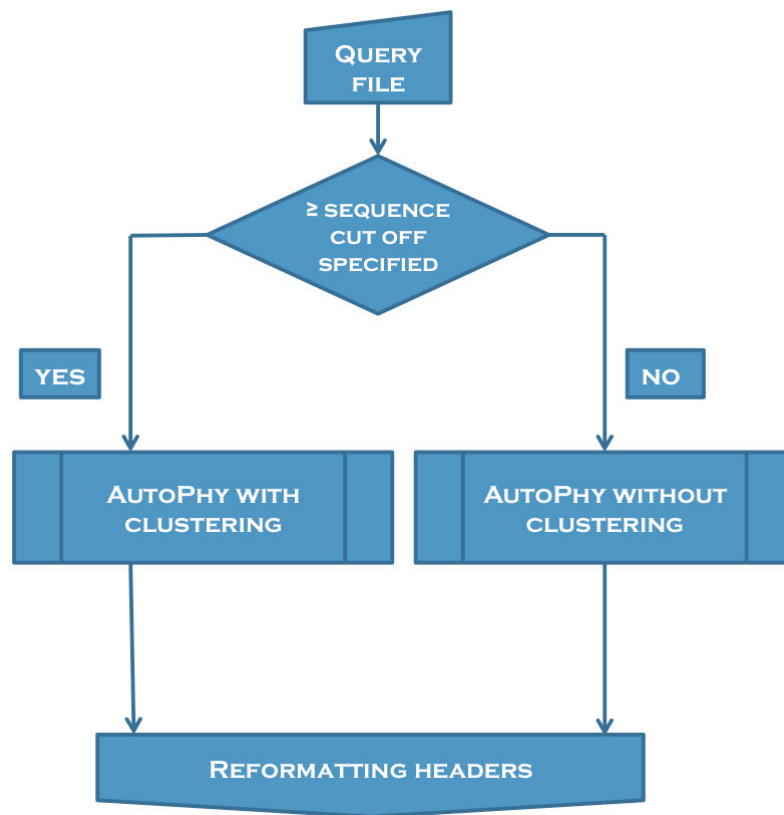


Figure 2.2: Describes the logic of selection of paths within the pipeline by the decision making script; helps AutoPhy take clustering or non-clustering path depending on size of dataset

There is a small script integrated into the wrapper that captures the last line in the file generated by AP-L-0, which lists the total number of sequences. If the number in this file is higher than the sequence cut off parameter, then AutoPhy chooses the pathway with the clustering option. If on the other hand, the number is lower than the cut off parameter mentioned, AutoPhy moves forward with the non-clustering option. The sequence cut off is an adjustable parameter. Tests have been carried out with different sequence cut offs to check for fallibility of the code in general and the decision code has responded as per the expected response. The default value of the sequence cut off is set at 500. If the value for the sequence cut off parameter is not defined, then the default value (500) is taken into consideration, depending on which the clustering or non-clustering option is chosen, using the logic code. The tool was developed for large datasets considering Virome receives metagenomic data, and more often than not the clustering step will be used.

The next step in the pipeline is the reformatting of headers in the query file. This step was taken for the sake of maintaining uniformity in the information of headers throughout the dataset. One of the first tools in the pipeline to truncate headers is MUSCLE [14] during multiple sequence alignment. The output is in phylip format, and this format has a 15-character limit when displaying the header. For reliable and understandable information about the tree once it is built, uniformity in header information is necessary.

The reformatting of headers in very simple terms is used to change the names of the headers to shorter ones. In this pipeline, numbers are substituted instead of the header names.

AP – 1 (autophy-01-query_sub_tab.pl) **and** **AP – 2** (autophy-02-query_head_sub.pl):

The header reformatting logic (Figure 2.3) was divided into two; AP – 1 and AP – 2.

AP – 1 assigns a number to each header in the query input (fasta file). This is stored in a tab delimited table format file can be accessed if necessary to check for an error. The second script, AP -2, accesses AP – 1 output and query file, and the original names in the query file are substituted with the designated numbers from AP – 1 output.

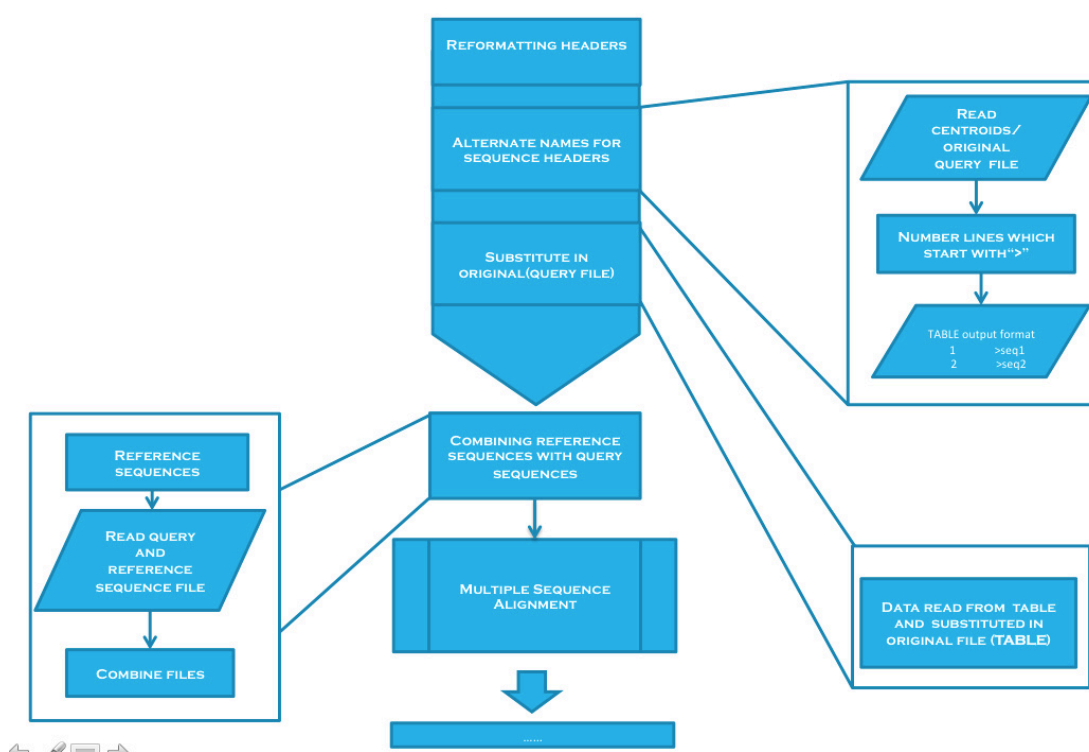


Figure 2.3: logical flow of data through of reformatting headers and combining of reference sequence file; AP – 1, AP – 2, AP – 5

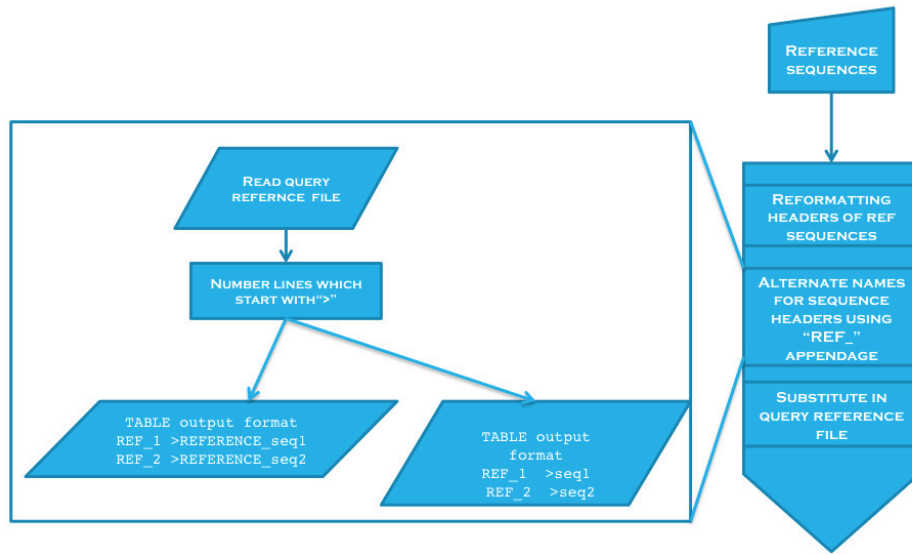


Figure 2.4: flowchart explaining the change in reference headers; AP – 3, AP – 4, AP – 5

AP – 3(autophy-03-ref_sub_tab.pl), **AP – 4**(autophy-04-ref_head_sub.pl), and **AP – 5**(autophy-03-1-ref_sub_tab.pl): Before header reformatting, there is an extra step of arranging the submitted reference sequences in decreasing order of length. This rearrangement of sequences within the file is done using the sort by length option offered by USEARCH. The reference sequence headers undergo a similar change to the input query headers (Figure 2.4). AP – 3 output is stored in a tab-delimited table. AP – 5 output is in the same format as AP – 3 output, but with a difference in the sequence header names. To be able to differentiate the query and reference sequences in the phylogenetic tree, AP – 5 output original reference sequences headers have a “REFERENCE_” appendage at the beginning of the original header name (shown in Figure 2.4) is added. AP – 5 output will be used in the last stage of the pipeline, AP –

12. AP – 4 logic works on the same principle as AP – 2, wherein AP – 3 output (e.g. REF_1), gets substituted instead of the original reference sequence header in the original reference query file (Figure 2.4).

AP – 6 (autophy-05-combine_ref_que.pl): The next script of importance combines AP – 4 output to AP – 2 output (left lower corner of Fig 2.3). Reference sequences (AP – 4 output) are combined with the query sequences (AP – 2 output) at the beginning of the file. The reason for combining the reference and query sequences is in aiding in optimized alignment. If the reference sequences (AP – 4 output), on the basis of which the query sequences are to be aligned are not added; query sequences will get forced into an alignment by the multiple sequence aligner, and in all likelihood this will form faulty alignments, ultimately producing imprecise and inferior quality phylogenetic trees.

AP – 6 output then undergoes multiple sequence alignment to produce output in the phylip format (Figure 2.5). The tool used to perform multiple sequence alignment is MUSCLE. MUSCLE is a widely used tool for multiple sequence alignment and is handy for large datasets. The tool is able to handle thousands of sequences while performing a multiple sequence alignment. Multiple sequence alignment is performed to align query sequences to the reference sequences. This helps in capturing the region of interest in the query sequences based on reference sequences.

```

5 121
87      MILTLDVENT VTERNGKMHL DPFE PDNTLV MVGMLTEAGD ETIVTFDHSE CAPTDNGRQI
88      MILVLDVENT VVERNGKMHL DPFE PENTLV MVGMLDEDGN EDIVTFDHAE HKPTLEGRSI
REF_10  MRHLNIDIET YSSNDIKNGV YKYADAEDFE ILLFAYSIDG GEVECLDLTR QSLPEDIKDM
89      MKITLDVENT VTHRDGKMHL DPFEVNNSLT MVGMLTDQDD ETLVVFDHEE AAPADQESFD
REF_9   MKINSLDIET EAVDPAEKLY AALQPWRLRQ GRSRITSIAV CRPGFTVDQI VNRGDNTLWL

      VQDMLDQTTL LVCHNASHDL VIDTANNLL KLVNKKWMLD FNVPLLEAK IGDNWLDTKS
      VQDKLDGTSL LICHNAAHDL VIDMANKNLL KIVNNKWQLD FNVPLLEAK IGPNWLDTKD
      LFDDKVRKHA FNAQFERVCL -----
      LVQSYLDEAT VLIMHNAHD  GFTSPFYMKD -----
      REMIDLLDSV GNDVVYAHNA -----

```

Figure 2.5: multiple sequence alignment in phylip format

Once the multiple sequence alignment output is obtained, the pipeline executes the Trim suite. The flowchart (Figure 2.6) in the next page summarizes this overall process that takes place within the suite.

AP – 7 (autophy-06-trim_reg_int.pl): Each one of the sequential steps in the trim suite (Figure 2.7) contributes in choosing the sequences with the maximum coverage of the length of the region of interest. The first script in the trim suite is the trimming script. This essential component in the suite is responsible for fragmenting the sequence to the specific part of interest, and has to be defined by the user. After obtaining the multiple sequence alignment, the aligned query sequences are then chopped to the same region as the region of interest in the reference sequence, using the coordinate input (as in if the region of interest is between position 1 and position 20 in the reference sequence, then all the query sequences are chopped at the same region in the alignment). The code takes in the start and end coordinate parameters (entered by user) and using the reference sequence header name (hard coded in AP – 7) to trim the aligned query sequences. The trimmed aligned sequences are then directed to another file with the header names.

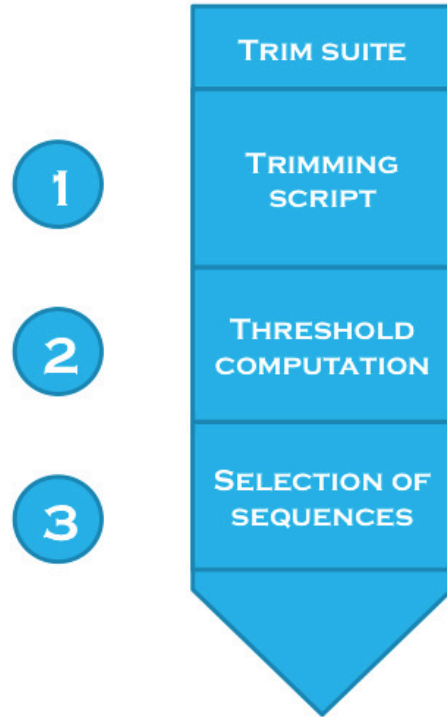


Figure 2.6: The flowchart shows the different processes that are a part of the Trim suite; 1. Trimming script - Chops the sequence to the required region keeping the reference sequence as a basis; 2. Script computes the ratio of the aligned region with respect to the cut off (Start and End coordinates) given; 3. Select the sequences – with a ratio above the threshold level.

AP – 7 output will be used to compute ratios for the next script. The ratio computation script plays an extremely important role in how the sequences are selected further downstream in the pipeline. This will be elucidated further (Figure 2.7).

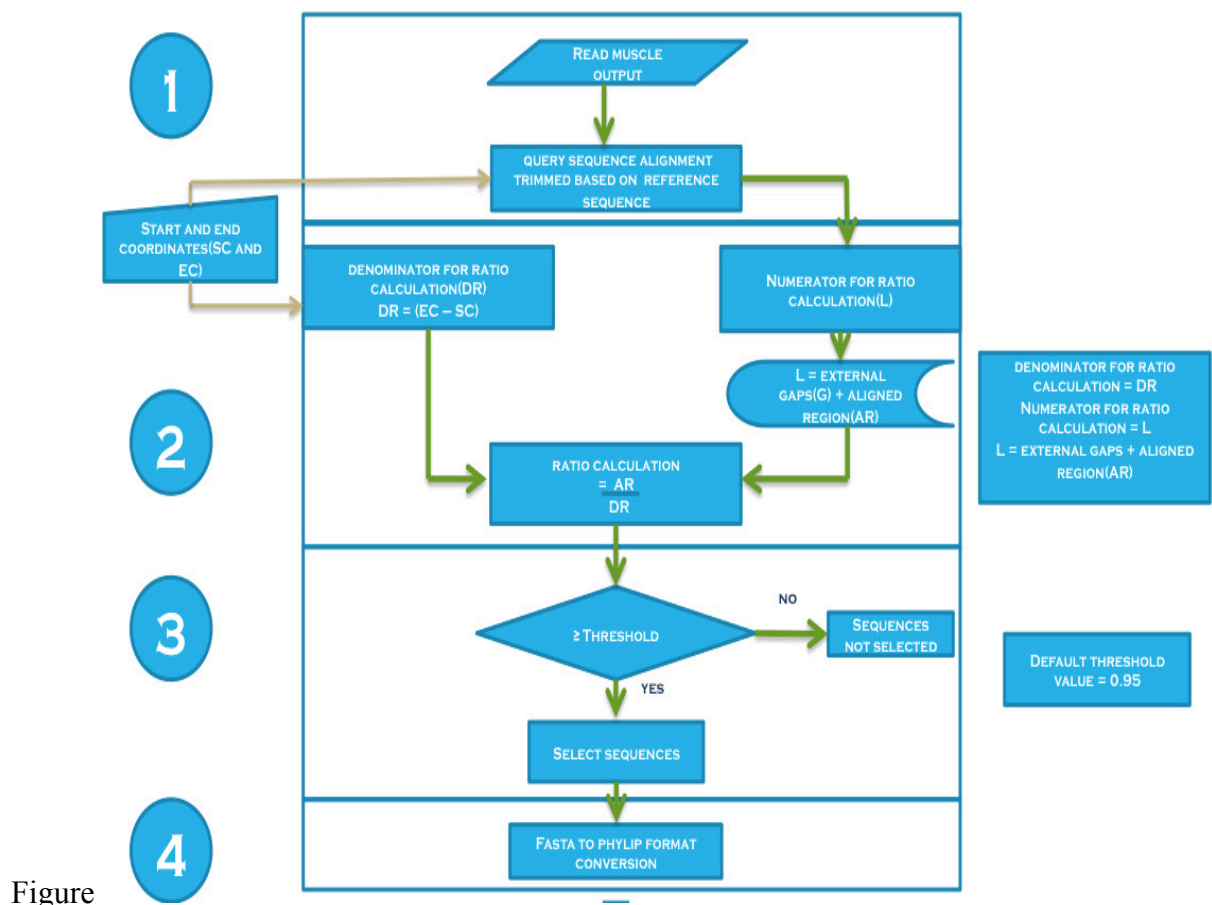


Figure 2.7: Describing logic of 1. trimming script flowing into, 2. calculates the threshold, moving onto 3. selects the sequences based on the threshold recorded for each sequence and 4. Converts everything back to the phylip format to be pushed further into the pipeline

AP – 8 (autophy-07-ratio_calc_for_selseqs.pl): The calculation of ratio is a necessary factor, required in the selection of sequences. The denominator of the ratio calculation is computed from the end and start coordinates submitted in AP – 7. The

numerator is calculated from the coverage of the length of query sequence, over the total length of the region of interest in the reference sequence. In the calculation of this coverage, the external gaps are not taken into consideration. This is further explained by way of an example (Figure 2.8). The specific region acquired from AP – 7 output quite often consists of two parts; the end gaps and the aligned amino acids including the internal gaps. This is explained further, below, in the form of an equation

E.g. Chopped sequence obtained from the trimming script:

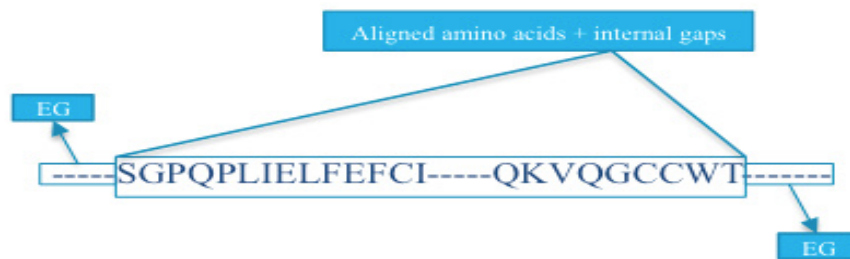


Figure 2.8: An example of a trimmed sequence; EG = external gaps

If, entire length of the chopped (trimmed) sequence (from Figure 2.8) = EG + (Aligned amino acids + internal gaps) + EG

$$\text{Then, ratio} = \frac{(\text{Aligned amino acids} + \text{internal gaps})}{(\text{End coordinate} + 1) - \text{Start coordinate}}$$

The ratio helps determining the percentage of coverage of the aligned region over the total length of the region of interest on the reference sequence

The threshold value is an adjustable parameter and can be determined based on how strict the criterion of the selection of sequences needs to be.

AP-9 (autophy-08-sel_seqs.pl): The default threshold value for AutoPhy has been set at 95%. This script selects sequences based on the threshold determined. Threshold is the percentage of coverage of the aligned amino acids including the internal gaps across the region of interest within the reference sequence provided. If the threshold chosen is higher, it increases the chances of:

1. Acquiring fully aligned regions to the region of interest in the reference sequence
2. Developing efficient phylogenetic trees with specific coverage

If the need is an absolute coverage of the region of interest in the reference sequence, the criterion will be a very strict 100%(or 1.000 in terms of ratio calculation), which has also been used as a part of the validation study. The sequences are selected and moved into a file in fasta format.

This is the point where the clustering and the non-clustering paths differ, in the pipeline. If clustering path is used, AP – 10 and AP – 11 are executed. If non-clustering path is used AP – 13 is executed.

Figure 2.9 shows the sequential steps that take place in the clustering path.

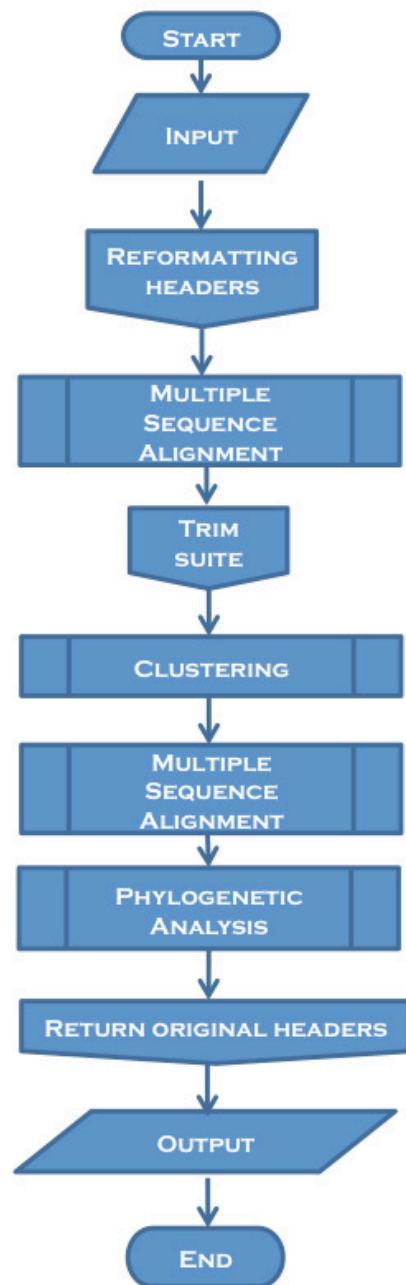


Figure 2.9: flowchart showing the flow of data in clustering path

USEARCH [15] processes fasta sequences. AP – 9 output (phylip format) obtained from the trim suite is converted to fasta format by removing the gaps that are introduced into the trimmed sequences as a part of the first multiple sequence alignment (Figure 2.9). The gaps are removed using a perl one liner to substitute “-”. USEARCH is the clustering tool used in the pipeline. In clustering, based on identity (sequence identity) of the query sequence to the representative sequence or centroid, the query sequence is put into the cluster with that particular centroid. Centroid is the sequence that represents the cluster, hence the name representative sequence. Clustering is performed on AP – 9 output, which was converted to fasta format. The sequence identity is an adjustable parameter. Default sequence identity value is at a 75%.

AP – 10 (autophy-choose_REF__seqs.pl) **and AP – 11** (autophy-13_combine_ref_que.pl): After clustering has been executed, the centroid sequences obtained as output need to undergo multiple sequence alignment before submission for phylogenetic analysis. The reference sequences are recombined to the beginning of the file in the centroid sequence file, using AP – 10 and AP – 11. AP – 10 and AP – 11 work the same way as AP – 6. The reference sequences are separated by AP – 10 and recombined at the beginning of the centroid sequence file by AP – 11.

If the non-clustering option is adopted (due to small dataset) after AP – 9, the AP – 9 output is submitted for phylogenetic analysis, directly. Before submission to phylogenetic analysis, the sequences need to undergo a format change.

AP – 13 (autophy-09-format_fa-phyl.pl): AP – 9 output is converted to phylip format and forwarded for phylogenetic analysis.

The steps followed by the pipeline after the small deviation to clustering or AP – 13 depending on clustering or non-clustering option (determined by step AP–L–0) chosen, the pipeline executes the next script, AP – 12.

AP-L-2 (autophy-log-2_sel_seqs_less.pl): Before execution of AP – 12, the issue of RAxML needs to be tackled. RAxML does not process the output (from step AP – 13 or AP – 11) if the number of sequences is less than five, in the query file being submitted for phylogenetic analysis. There is a hard coded logic for the stoppage of the pipeline in case this scenario rises. If the file has more than five sequences then the pipeline continues to completion with the implementation of RAxML. AP–L–2 works on the same logic as AP–L–0.

RAxML or Randomized Axelerated Maximum Likelihood is the tool being used for phylogenetic analysis and producing the associated analyzed newick file. It produces a result with the best tree in the newick format and this file is one of the inputs for AP – 12.

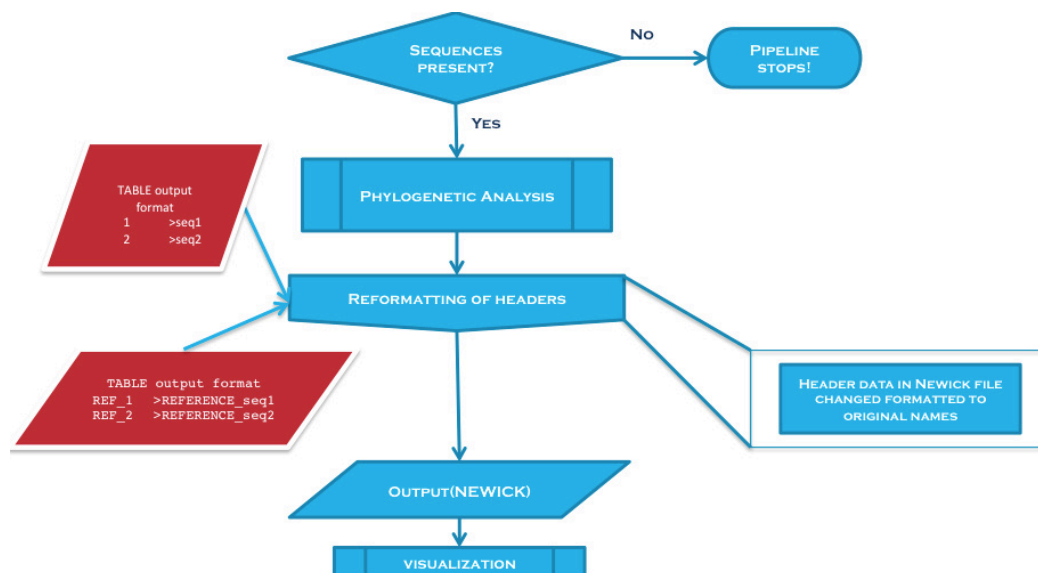


Figure 2.10: conceptual flowchart of the data from the trim suite to the rest of the pipeline

AP – 12(autophy-10-header_returner.pl): AP – 12 is executed on the best tree file obtained after phylogenetic analysis, in the newick format. Output of AP – 1, AP – 5 and RAxML best tree file are the input (Figure 2.10) for this script. Using this script the original header names are returned to the newick file. This file with the original header names can then be visualized using visualization software. The software used for this purpose during the development of this tool was, FigTree or Archaeopteryx.

Chapter 3

VALIDATION FOR AUTOPHY

3.1. Part – 1: Validation of codes in the pipeline

The validation of scripts in Part -1 are meant to check the functionality, format, and various criteria based on which the codes are executed, and this should give a better understanding on the inner workings of the codes and results obtained.

The dataset size will vary in each section, depending on the logic that needs to be validated. Each section of Part -1, will contain this basic information on the codes used:

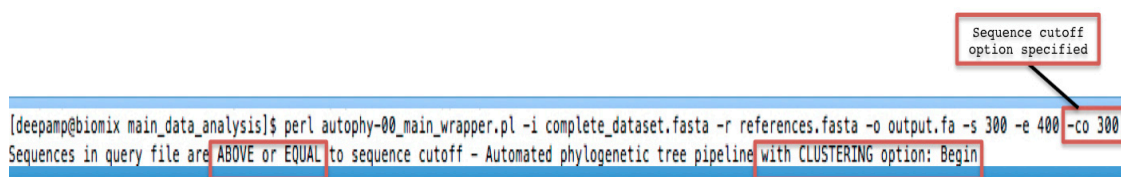
- I. Function of code
- II. Format of input file
- III. Expected output
- IV. Specific parts of code in cases of regular expressions or calculations performed
- V. Criteria tested on code (if applied)
- VI. Results of each criteria applied to code (if applied)
- VII. Format of output file

AP-L-0: Validating code - autophy-log-1_seqs_above_cutoff.pl

Function of code: this code was written to calculate the number of sequences in the query file based on the sequence cutoff, submitted to AutoPhy. The code produces an output file showing the number of lines within the query file. The total of this is matched with the sequence cut off specified. If this number is greater

than equal to the number specified by the cutoff variable, then clustering option is adopted. If not, the non-clustering option is adopted.

Below, this cutoff option consideration is taken into account and demonstrated for a random start region of 300 with an end region of 400 and cut off at 300. In this example, a file with more than 1000 sequences has been used to show the cutoff variable usability.

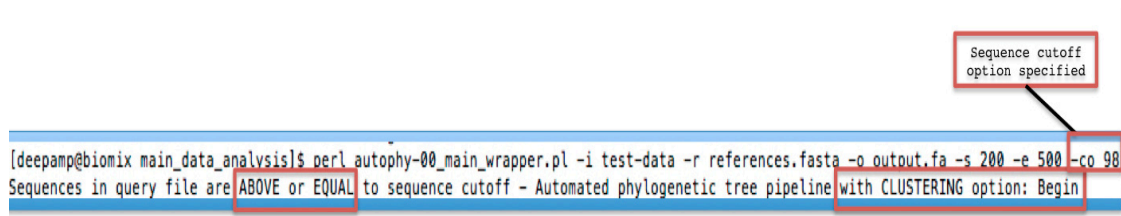


```
[deepamp@biomix main_data_analysis]$ perl autophy-00_main_wrapper.pl -i complete_dataset.fasta -r references.fasta -o output.fa -s 300 -e 400 -co 300
Sequences in query file are ABOVE or EQUAL to sequence cutoff - Automated phylogenetic tree pipeline with CLUSTERING option: Begin
```

Sequence cutoff option specified

Figure 3.1: cutoff parameter applied within logic code

Below, this cutoff option consideration is taken into account and demonstrated for a random start region of 200 with an end region of 400 and cutoff at 98 sequences. This dataset consisted of a 100 sequences. The cut off value was randomly chosen to demonstrate, adaptability of AutoPhy.



```
[deepamp@biomix main_data_analysis]$ perl autophy-00_main_wrapper.pl -i test-data -r references.fasta -o output.fa -s 200 -e 500 -co 98
Sequences in query file are ABOVE or EQUAL to sequence cutoff - Automated phylogenetic tree pipeline with CLUSTERING option: Begin
```

Sequence cutoff option specified

Figure 3.2: cutoff parameter applied within logic code

Below, the cutoff option is not specified and it is seen that the AutoPhy with the non-clustering option is adopted. When the sequence cutoff option is not specified, AutoPhy takes the default sequence cutoff into account and adopts the path of the non-clustering option. Here, sequences with random cut offs with a dataset consisting of 100 sequences.

These examples show how if cut off variable is not mentioned, AutoPhy smoothly takes over another path, without the clustering option to process the data.

Below, this cutoff option consideration is taken into account and demonstrated for a random start region of 300 with an end region of 400 with sequence cutoff variable not mentioned.

```
[deepamp@biomix main_data_analysis]$ perl autophy-00_main_wrapper.pl -i test-data -r references.fasta -o output.fa -s 300 -e 400  
Sequences in query file are BELOW sequence cutoff - Automated phylogenetic tree pipeline without CLUSTERING option: Begin
```

Figure 3.3: cutoff parameter not applied not utilized as the number of sequences was below the cutoff variable

Below, this cutoff option consideration is taken into account and demonstrated for a random start region of 200 with an end region of 500 with sequence cutoff variable not mentioned.

```
[deepamp@biomix main_data_analysis]$ perl autophy-00_main_wrapper.pl -i test-data -r references.fasta -o output.fa -s 200 -e 500  
Sequences in query file are BELOW sequence cutoff - Automated phylogenetic tree pipeline without CLUSTERING option: Begin
```

Figure 3.4: cutoff parameter not applied as the number of sequences was below the cutoff variable

Steps AP – 1 to AP – 5, a set of 5 sequences has been taken to show how the codes function.

AP – 1: Validating code – autophy-01-query_sub_tab.pl

Function of code: The main function of this code is to create a output file in a table format listing out the header and the number associated with it. The number gets substituted in the output of the next data file.

Table 3.1: Input and output files for AP – 1

Input file	
<pre> >DTF_L MVVHYAYS DGLDVRSIVNGYRSGEADFHEMVAEIAQISRRQAKTINLGMMYGMKGKLMNELGIDKEEAEEIVSIYQNKV PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFFEPRKFGEKGFYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT KKAMVDLYEEDGIIPHIQVHDELNISVENKGEALSIIKKM >EC_L LPNLQRTINDVLDNESLIPAVRAVLEARIGATKITSQVQRAIGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPNLSRG IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGDDGMLTVMDYSQIEVRVLAWLAGQQSVLDAFEAGQDIYVNMAAKVF GENEVSDLQRNTVKGKPLILGCGFGLGSKTFEIFG >EC_F KNPTFLLTYGGTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYTTVAFGLRVRTPIAKTILGNRAT PYEAQSEARTMGNAHGQSYGMLNNRAAIELQERLFDISKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNLIECMEWQDN DNIRHEQVKLGQLSIFYPSWDKEYKL >USR_Y MLNNWLDNLKQDGRHLHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYGKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQRVDDT >ABF_L MPDSMRHPVVDTIVEEYHKGVDVLHQMVAADLANIKRKEAKTVNLGIMYGMGVGKLAQLDISKEEAKNLTIEQHRTNVPFV KQLASIASQRAEDQGQIRTLGRKCRFHLWEPKTFGYNKPMLREAAKKEYGNINNLKRAFTYKALNKLIQGSAAD </pre>	
Expected output	
1	>DTF_L
2	>EC_L
3	>EC_F
4	>USR_Y
5	>ABF_L
Actual output	

```
[deepamp@biomix section-1]$ perl autophy-01-query_sub_tab.pl
1      >DTF_L
2      >EC_L
3      >EC_F
4      >USR_Y
5      >ABF_L
```

A separate output file is created wherein the header names are extracted from the main query file and given numbers.

AP – 2: Validating code – autophy-02-query_head_sub.pl

Function of code: The code uses the table produced by AP – 1 and substitutes the header names with the associated number in the output file

Table 3.2: Input and output files for AP – 2

Input file – 1	
>DTF_L MVVHYAYS DGLD VRSIVNGYRSGEADFHEMVAEIAQISRRQAKTINLGMMYGMGKGLMNELGIDKEEAEEIVSIYQNKV PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFFEPRKFGEKGFYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT KKAMVDLYEEDGIIPIHIQVHDELNISVENKGEALSIIKKM >EC_L LPNLQRGTINDVLDNESLIPAVRAVLEARIGATKITSAKVQRALGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPQNLSRG IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGDDGMLTVMDSQIEVRVLAWLAGQQSVLDAFEAGQDIYVNMAAKVF GENEVSDLQRNTVKGKPLILGCGFGLGSKTFEIFG >EC_F KNPTFLLTYGGTTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYYTVAFGLRV RTPILAKTILGNRAT PYEAQSEARTMGNAHQSYGMLNNRAAIELQERLFD SKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNLIECMEWQDN DNIRHEQVKLGGLSIFYPSWDKEYKL >USR_Y MLNNWLDNLKQDGRHLHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGSSDDGRTLKARFLSNTPALAGLRQRVDDT >ABF_L MPDSMRHPVVD TIVEEYHKGDVDLHQMVADLANIKRKEAKTVNLGIMYGMGVGKLAAQLDISKEEAKNLIEQHRTNVPFV KQLASIASQRAEDQGQIRTL LGRKCRFHLWE PKTFGYNKPMRLEEAKKEYGNINNLKRAFTYKALNKLIQGSAAD	
Input file – 2	
1 >DTF_L 2 >EC_L 3 >EC_F 4 >USR_Y 5 >ABF_L	
Expected Output	

```

>1
MNVHYAYS DGLDVRSIVNGYRSGEAD FHEMVAEIAQISRRQAKTINLGMMYGMKGKLMNELGIDKEEAEEIVSIYQNKV
PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFEPKFGKGFYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT
KKAMVDLYEEDGIIPIHQVHDELNISVENKGEALSIIKKM
>2
LPNLQRGTINDVLNDSLIPAVRAVLEARIGATKITSAKVQRALGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPQNLSRG
IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGDDGMLTVMDSQIEVRVLAWLAGQSVLDAFEAGQDIYVNMAAKVF
GENEVDLQRNTVGKPLILGCGFGLGSKTFEIFG
>3
KNPTFLLTYGGTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYYTTFVAFGLRV RTPILAKTILGNRAT
PYEAQSEARTMGNAHGQSYGMLNNRAAIELQERLFD SKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNIECMEWQDN
DNIRHEQVKLGGLSIFYPWDKEYKL
>4
MLNNWLDNLKQDGR LHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYGKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN
DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQRVDDT
>5
MPDSMRHPVVDTIVEEYHKG DVDLHQM VADLANIKRKEAKTVNLGIMYGMVGKLAQLDISKEEAKN LIEQHRTNVPFV
KQLASIASQRAEDQGQIR TLLGRKCRFHLWE PKTFGYNKPMRLEEAKKEYGNINNLKRAFTYKALNKLIQGSAAD

```

Actual output

```

[deepamp@biomix section-1]$ perl autophy-02-query_head_sub.pl
>1
MNVHYAYS DGLDVRSIVNGYRSGEAD FHEMVAEIAQISRRQAKTINLGMMYGMKGKLMNELGIDKEEAEEIVSIYQNKV
PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFEPKFGKGFYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT
KKAMVDLYEEDGIIPIHQVHDELNISVENKGEALSIIKKM
>2
LPNLQRGTINDVLNDSLIPAVRAVLEARIGATKITSAKVQRALGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPQNLSRG
IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGDDGMLTVMDSQIEVRVLAWLAGQSVLDAFEAGQDIYVNMAAKVF
GENEVDLQRNTVGKPLILGCGFGLGSKTFEIFG
>3
KNPTFLLTYGGTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYYTTFVAFGLRV RTPILAKTILGNRAT
PYEAQSEARTMGNAHGQSYGMLNNRAAIELQERLFD SKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNIECMEWQDN
DNIRHEQVKLGGLSIFYPWDKEYKL
>4
MLNNWLDNLKQDGR LHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYGKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN
DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQRVDDT
>5
MPDSMRHPVVDTIVEEYHKG DVDLHQM VADLANIKRKEAKTVNLGIMYGMVGKLAQLDISKEEAKN LIEQHRTNVPFV
KQLASIASQRAEDQGQIR TLLGRKCRFHLWE PKTFGYNKPMRLEEAKKEYGNINNLKRAFTYKALNKLIQGSAAD

```

AP – 3: Validating code – autophy-03-ref_sub_tab.pl

Function of code: This code works on the same principle as AP – 1, but uses a different input file. The input file it uses is the results of a file with reference sequences that have been sorted by length using the USEARCH command. This helps in arranging the sequences in increasing order for length. This helps in better alignment and coverage of the region of interest within the reference sequence.

The sequences with the substituted names are appended with a “REF_” to be able to differentiate them from the query sequences. The reference sequences have not been displayed in the input file section completely for easier an easier view

Table 3.3: Input and output files for AP – 3

Input file
<pre>>Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] MVQIPQNPLILVDGSSYLRYAHAFPLTNSAGEPTGAMYGVNLMSLIMQYKPTHAAVVFDAKGKTRDELFEHYKSHRPPMPD DLRAQIEPLHAMVKAMGPLLLAVSGVEADDDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGVP PELIIDFLALMGDSSDNIPGVPGVGEKTAQALLQGLGGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL ELTCEQLEVQPPAAEELLGLFKKYEFKRWTADVEAGKWLQAKGVKPAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA KLEKAPVFAFDTETDSDLNISANLVGLSFAIEPGVAAYIPV >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase MKTLDNIETYSDDLTKVGVYKADSPNFEILLFAYSVDGQPVCECDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYLG VPYYLDPAQWHCTMVHANELGLPASLGQCAKYLNIEQQKDRGTQQLINFFSKPKCKPTKKNMGRTRNLPEHAPEKWQTFIEYCIQDV NVEMAIANKLNRFPVPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNLSLAQLKKWLEEQGTPEE KLGKEVVLKALALGNLPENVAEVLKLRSLSNSSTKKYLMMDNARCSN >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFDHAFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCIQDV VEVEMTIAHKIKDFPVTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLDKQSKKEELLNQAKHITGLENPNPSPTQLLAWLKDDQGLD IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMMDMM >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFDHAFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCIQDV >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A domain MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFDHAFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQSKDKAGKNLIRY</pre>
Expected output
<pre>REF_1 >Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] REF_2 >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase REF_3 >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase REF_4 >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase</pre>

REF_5 >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A
Actual output
<pre>[deepamp@biomix section-1]\$ perl autophy-03-ref_sub_tab.pl REF_1 >Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] REF_2 >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase REF_3 >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase REF_4 >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase REF_5 >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A domain</pre>

AP – 4: Validating code – autophy-04-ref_head_sub.pl

Function of code: The function of this code is to use the query file and output table file from AP – 2 to substitute the header names with the substituted names, and create a new output file shown below, and used as input for the next script.

Table 3.4: Input and output files for AP – 4

Input file – 1
<pre>>Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] MVQIPQNPLILVDGSSYLYRAYHAFPLTNSAGEPTGAMYGVNMLRSLIMQYKPTHAAVVFDAKGKTFRDELFEHYKSHRPPMPD DLRAQIEPLHAMVKAMGLPLLAVSGVEADDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGVP PELIIDFLALMGDSSDNIPGVPGVGEKTAQALLQQLGGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKVEVAYLSYQLATIKTDVEL ELTCEQLEVPQPAEELLGLFKKYEFKRWTADEAGKWLQAGVKPAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA KLEKAPVFAFDTTETDSDLNISANLVGLSFAIEPGVAAYIPV >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase MKTLNIDIETYSDEDLTKVGVIKYADSPNFEILLFAYSVDGQPVCECDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYLG VPYYLDPAQWHCTMVHANELGLPASLGQCAKYLNIEQQKDTRGTLINFFSKPCKPTKKNMTRNLPEHAPEKWQTFIEYCIQDV NVEMAIANKLNRFPVPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNSLAQLKKWLEEQGTPE KLGKEVVLKALALGNLPENVAEVLKRLSLSNSSTKKYLMMDNARCSN >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQNKDKAGKNLIRYFSIPCKPTKVNGGRTNRLPEHDLEKWQQFIDYCIQDV VEVEMTIAHKIKDFPVTAEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQKEELLNQAQKHTGLENPNSPQQLLAWLKDDQGLD IPNLQKKTQVEYLKEATGKAKKMLEIRLQMSKTSVKKYNKMHDDMM >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQNKDKAGKNLIRYFSIPCKPTKVNGGRTNRLPEHDLEKWQQFIDYCIQDV >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A domain MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQNKDKAGKNLIRY</pre>
Input file – 2
<pre>REF_1 >Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] REF_2 >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase</pre>

REF_3 >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase
REF_4 >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase
REF_5 >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A 1

Expected output

```
>REF_1
MVQIPQNPLILVDGSSYLRYAYHAFPLTNSAGEPTGAMYGVNLMSLRSLIMQYKPTHAAVVFDAKGKTRFDELFEHYKSHRPPMPD
DLRAQIEPLHAMVKAMGLPLLAVSGVEADDDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGV
PELIIDFLALMGDSSDNIPGVPGVGKTAQALLQGLGGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL
ELTCEQLEVQPPAAEELLGLFKKYEFKRWTADEAGKWLQAGVKPKAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA
KLEKAPVFAFDTEDSLNDNISANLVGLSFAIEPGVAAYIPV

>REF_2
MKTINIDIETYSDEDLTKVGVIYKADSPNFEILLFAYSVDGQPVCEEDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYL
VPPYLDPAQWHCTMVHANELGLPASLQGCAYKLNIEQQKDRGTQQLINFFSKPKCKPTKKNGMTRNRLPEHAPEKWQTFIEYCIQDV
NVEMAIANKLNRFPVPPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNSLAQLKKWLEEQGTPEF
KLGEVVLKALALGNLPENVAEVLKLRLSLSNSSTKKYLMMDNARCSN

>REF_3
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADFETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNRLPEHDLEKWQQFIDYCI
RDVEVEMTIAHKIKDPVPTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQSKKEELLNQAKHITGLENPNSPTQLLAWLKDDQGL
IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMMDMMC

>REF_4
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADFETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNRLPEHDLEKWQQFIDYCI
RDVEVEMTIAHKIKDPVPTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQSKKEELLNQAKHITGLENPNSPTQLLAWLKDDQGL
IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMMDMMC

>REF_5
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADYETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQSKDKAGKNLIRY
```

Actual output

```
[deepan@bionix section-1]$ perl autophy-04-ref_head_sub.pl
>REF_1
MVQIPQNPLILVDGSSYLRYAYHAFPLTNSAGEPTGAMYGVNLMSLRSLIMQYKPTHAAVVFDAKGKTRFDELFEHYKSHRPPMPD
DLRAQIEPLHAMVKAMGLPLLAVSGVEADDDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGV
PELIIDFLALMGDSSDNIPGVPGVGKTAQALLQGLGGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL
ELTCEQLEVQPPAAEELLGLFKKYEFKRWTADEAGKWLQAGVKPKAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA
KLEKAPVFAFDTEDSLNDNISANLVGLSFAIEPGVAAYIPV

>REF_2
MKTINIDIETYSDEDLTKVGVIYKADSPNFEILLFAYSVDGQPVCEEDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYL
VPPYLDPAQWHCTMVHANELGLPASLQGCAYKLNIEQQKDRGTQQLINFFSKPKCKPTKKNGMTRNRLPEHAPEKWQTFIEYCIQDV
NVEMAIANKLNRFPVPPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNSLAQLKKWLEEQGTPEF
KLGEVVLKALALGNLPENVAEVLKLRLSLSNSSTKKYLMMDNARCSN

>REF_3
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADFETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNRLPEHDLEKWQQFIDYCI
RDVEVEMTIAHKIKDPVPTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQSKKEELLNQAKHITGLENPNSPTQLLAWLKDDQGL
IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMMDMMC

>REF_4
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADFETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNRLPEHDLEKWQQFIDYCI
RDVEVEMTIAHKIKDPVPTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQSKKEELLNQAKHITGLENPNSPTQLLAWLKDDQGL
IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMMDMMC

>REF_5
MNIDIETYSNDISKGAYKYTEAEDEFELIIAYSIDGGAISAIMTKVDNEPFPHADYETFKIALFDPAVKKYAFNANFERTCLAK
HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQSKDKAGKNLIRY
```

AP – 5: Validating code – autophy-03-1-ref_sub_tab.pl

Function of code: The code works on the same principle as AP – 3, and creates a table with a different appendage. The output file with the table gets created and is used in the last step of the pipeline. The original header names have an appendage of “REFERENCE_”. This code has been added to make the reference sequences more obvious in node labels on a phylogenetic tree.

Table 3.5: Input and output files for AP – 5

Input file
<pre>>Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] MVQIPQNPLILVDGSSYLRYAHAFPLTNSAGEPTGAMYGVNLMLRSLIMQYKPTHAAVVFDAKGKTRDELFEHYKSHRPPMPD DLRAQIEPLHAMVKAMGLPLLAVSGVEADDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGVP PELIIDFLALMGDSSDNIPGVPGVGKTAQALLQGLGGLDLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL ELTCEQLEVQPPAAEELLGLFKKYEFKRWTADVEAGKWLQAKGVKPAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA KLEKAPVFAFDTEETDSDLNISANLVGLSFAIEPGVAAYIPV >Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase MKTNLNIDIETYSDDLTKVGKYADSPNFEILLFAYSVDGQPVCECDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYLG VPYYLDPAQWHCTMVHANELGLPASLGQCAKYLNIEQQKDRGTQLINFFSKPCKPTKKNGMRTRNLPEHAPEKWQTFIEYCIQDV NVEMAIAANKLNRFPVPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNSLAQLKKWLEEQGTPEE KLGKEVVLKALALGNLPENVAEVLKLRSLSNSSTKKYLMMDNARCSN >Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase MNIDIETYSSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIDMTKVDNEPFDHADFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTNRLPEHDLEKWQQFIDYCIRD VEVEMTIAHKIKDFPVTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLDDKQSKHEELLNQAKHITGLENPNPSPTQLLAWLKDDQGLD IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYKNMHDMMC >Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase MNIDIETYSSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIDMTKVDNEPFDHADFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQNKDKAGKNLIRYFSIPCKPTKVNGGRTNRLPEHDLEKWQQFIDYCIRD >Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A domain MNIDIETYSSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIDMTKVDNEPFDHADFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRQLQSKDKAGKNLIRY</pre>
Expected Output
<pre>REF_1 >REFERENCE_Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39] REF_2 >REFERENCE_Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase REF_3 >REFERENCE_Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase REF_4 >REFERENCE_Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase REF_5 >REFERENCE_Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A</pre>
Actual output

```
[deepamp@biomix section-1]$ perl autophy-03-1-ref_sub_tab.pl
REF_1 >REFERENCE_Escheria_coli_IAI39 DNA polymerase I [Escherichia coli IAI39]
REF_2 >REFERENCE_Enterococcus_phage_phiFL4A_L phage phiFL4A => DNA polymerase
REF_3 >REFERENCE_Staphylococcus_phage_phi2958PVL_L phage phi2958PVL => DNA polymerase
REF_4 >REFERENCE_Staphylococcus_phage_SMSAP5_L phage SMSAP5 => DNA polymerase
REF_5 >REFERENCE_Staphylococcus_prophage_phi_12_L prophage phi 12 => DNA polymerase A domain
```

AP – 6: Validating code – autophy-05-combine_ref_que.pl

Function of code: The reference sequences are integrated with the query sequences, to help with downstream analysis. The sequences are added at the beginning of the file so that it helps with better alignment in the multiple sequence alignment.

Table 3.6: Input and output files for AP – 6

Input file – 1
<pre>>REF_1 MVQIPQNPLILVDGSSYLYRAYHAFPLTNSAGEPTGAMYGVNLMLRSLIMQYKPTHAAVVFDAKGKTFRDELFEHYKSHRPPMPD DLRAQIEPLHAMVKAMGLPLLAVSGVEADDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVNKGYP PELIIDFLALMGDSSDNIPGVPGVGEKTAQALLQGLGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL ELTCEQLEVQPPAAEELLGLFKKYEFKRWTADVEAGKWLQAKGVKPAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA KLEKAPVFAFDTTETDSDLNISANLVGLSFAIEPGVAAYIPV >REF_2 MKTLNIDIETYSDEDLTKVGVIYADSPNFEILLFAYSVDGQPVCECDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYLG VPYYLDPAQWHCTMVHANELGLPASLGQCAKYLNIEQQKDTRGTLINFFSKPCKPTKKNMTRNLPEHAPEKWQTFIEYCIQDV NVEMAIANKLNRFPVPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNLSAQLKKWLEEQGTPE KLGEVVLKALALGNLPENVAEVLKRLSLSNSSTKKYLMMDNARCSN >REF_3 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQNQKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCIQDV VEVEMTIAHKIKDFPVTAEQAYWVFDQHINDRGIKLSKSLMLGANVLQKSKKEELLNQAKHITGLENPNSPQLLAWLKDDQGLD IPNLQKKTQVEYLKEATGKAKKMLEIRLQMSKTSVKKYNMHDMMC >REF_4 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQNQKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCIQDV >REF_5 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIIDMTKVDNEPFDHADFETFKIALFDPAVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGVLRLQSKDKAGKNLIRY</pre>
Input file – 2

<p>>1 MVVHYAYS DGLD VRSIVNGYRSGEAD FHEMVAEIAQISRRQAKTINLGMMYGMKGKLMNELGIDKEEAEEIVSIYQNKV PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFEPRKFGEKGYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT KKAMVDLYEEDGIIPHIQVHDELNISVENKGEALSIIKKKM</p> <p>>2 LPNLQRGTINDVLDNESLIPAVRAVLEARIGATKITSAKVQRALGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPQNLSRG IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGGDGGMLTVMDSQIEVRVLAWLAGQQSVLDAFEAGQDIYVNMAAKVF GENEVSDLQRNTVGKPLILGCGFGLGSKTFEIFG</p> <p>>3 KNPTFLLTYGGTTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYYTTFVAFGLRVRTPIAKTILGNRAT PYEAQSEARTMGNAHQSYGMLNNRAAIELQERLFDISKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNLIECMEWQDN DNIRHEQVKLGQLSIFYPSWDKEYKL</p> <p>>4 MLNNWLDNLKQDGRHLHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQRVDDT</p> <p>>5 MPDSMRHPVVDTIVEEYHKGDVDLHQMVAADLANIKRKEAKTVNLGIMYGMGVGKLAAQLDISKEEAKNLIEQHRTNVFPV KQLASIASQRAEDQGQIRITLLGRKCRFHLWEPKTFGYNKPMRLEEAKKEYGNINNLKRAFTYKALNKLIQGSAAD</p>
<p>Expected output</p>
<p>>REF_1 MVQIPQNPLILVDGSSSYLYRAYHAFPLTNSAGEPTGAMYGVNLMLRSLIMQYKPTHAAVVFDAGKGTFRDELFEHYKSHRPPMPD DLRAQIEPLHAMVKAMGLPLLAVSGVEADDVIGTLAREAEKAGRPVLISTGDKDMAQLVTPNITLINTMTNTILGPEEVVNKYGVP PELIIDFLALMGDSSDNIPGVPGVGEKTAQALLQGLGGLDTLYAEPEKIAGLSFRGAKTMAAKLEQNKEVAYLSYQLATIKTDVEL ELTCEQLEVPAAEELLGLFKKYEFKRWTADEAGKWLQAKGVKPAARPQETSVADEAPEVTATVISYDNYVTILDEETLKEWIA KLEKAPVFAFDTFETDSDLNISANLVGLSFAIEPGVAAYIPV</p> <p>>REF_2 MKTLDNIETYSDEDLTKVGVIYADSPNFEILLFAYSVDGQPVCEDLTISEIPDEIVAALTDKNVLKIAFNAQFERVCLSKYL VPPYLDPAQWHCTMVHANELGLPASLGGQCAKYLNIEQQKDRGTQQLINFFSKPCKPTKKNMTRNLPEHAPEKWQTFIEYCIQDV NVEMAIANKLNRFPVPPESEWKLYTLDQRINDRGAEIDHELATAAIDIMADLSEAGLNEMKELTGLENPNSLAQLKKWLEEQGTPFE KLGKEVVLKALALGNLPENVAEVLKLRSLSNSSTKKYLMMDNARCSN</p> <p>>REF_3 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFHADFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRLLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCI VEVEMTIAHKIKDFPVTAIEQAYWVFDQHINDRGIKLSKSLMLGANVLQKQSKHEELLNQAKHITGLENPNSTQLLAWLKDDQGLD IPNLQKKTVQEYLKEATGKAKKMLEIRLQMSKTSVKKYNKMHMDC</p> <p>>REF_4 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFHADFETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRLLQNKDKAGKNLIRYFSIPCKPTKVNGGRTRNLPEHDLEKWQQFIDYCI >REF_5 MNIDIETYSNDISKCGAYKYTEAEDFEILIIAYSIDGGAISAIMTKVDNEPFHADYETFKIALFDPVKKYAFNANFERTCLAK HFNKQMPPEEWICTMVNSMRIGLPASLDKVGEVLRLLQSKDKAGKNLIRY</p> <p>>1 MVVHYAYS DGLD VRSIVNGYRSGEAD FHEMVAEIAQISRRQAKTINLGMMYGMKGKLMNELGIDKEEAEEIVSIYQNKV PFVKQLTYNVMDKASARGEIKTLLGRHCRFPFFEPRKFGEKGYKTKEEAIDALGHGNYKRAGTYKALNKLIQGSAADQT KKAMVDLYEEDGIIPHIQVHDELNISVENKGEALSIIKKKM</p> <p>>2 LPNLQRGTINDVLDNESLIPAVRAVLEARIGATKITSAKVQRALGLVAGDGRIRNCLAYHGAHTGRWSGRSFQPQNLSRG IKCDVDQLVAETMAGYQPTDDELSTLIRACVVGGDGGMLTVMDSQIEVRVLAWLAGQQSVLDAFEAGQDIYVNMAAKVF GENEVSDLQRNTVGKPLILGCGFGLGSKTFEIFG</p> <p>>3 KNPTFLLTYGGTTYGLIKLGFSEIEAKAIEDSYHALYKVSDDWIRSKVQEASKTGYYTTFVAFGLRVRTPIAKTILGNRAT PYEAQSEARTMGNAHQSYGMLNNRAAIELQERLFDISKYVYDILPISHIHDAQYFLVRNKAGAVKWLNDNLIECMEWQDN DNIRHEQVKLGQLSIFYPSWDKEYKL</p> <p>>4 MLNNWLDNLKQDGRHLHGRVNTNGAITGRMTHSDPNLAQVPAGYSPYKEMRSLFTIPNGYKLVGADAAQLELRMLAHYMN DKDYTNEILNGDVHTANQIAAGLDTRDKAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQRVDDT</p> <p>>5 MPDSMRHPVVDTIVEEYHKGDVDLHQMVAADLANIKRKEAKTVNLGIMYGMGVGKLAAQLDISKEEAKNLIEQHRTNVFPV KQLASIASQRAEDQGQIRITLLGRKCRFHLWEPKTFGYNKPMRLEEAKKEYGNINNLKRAFTYKALNKLIQGSAAD</p>
<p>Actual output</p>

```

[deepamp@biomix section-1]$ perl autophy-05-combine_ref_que.pl
>REF_1
MVIQIPNPLILVGGSSYLRYAYHAPPLNSAGEPTGAMYGVLMRLSLIMQYKTHAAVFDAGKXTFDELFEHYKSHRPPMDQRLAEPLHAWKMGPLLAIVSGVEADVIGTLAREAEKAGRPVLISTGQKMAQVLTWNTLINTNTILGPEEVNKYGVPELIIIDLALM
GDSOINIPGVPGVGEKTAQALLQGLGDLTYAEPEKIAGLSFRGAKTMAAKLEQNKVAYLSYQATIKTDVELELTCEQLLEVOPAAEELGLFKKYEKRWADVEAGKWLQAGVKPAARPQETSVADEAPEVATVTSYDNYVILDEETLKEWIAKLEKAPVAFOTETSDNLSA
NLVGLSFAIEQPGVAAVYIPVAHDYLDAPQISRRERALELLKPLLEDEKALKVGNLYDRGLANYGIELRGIAFDTHLESYILNSVAGRHMDSLAERMLKHKTITFEEIAGKGNQLTNQIALEAGRYAAEDAVTLQLHLXMPDLQKHGKPLNVFENIEMPLVPLSRIEERNGVKIDP
KVLHNSHEELTLRLAELEKKAIEAGEEPNLSSTKQLQTLFEKQGTGKPLKTPGGAPSTSEEVLEELALDYLPLKVILYRGLAKLSTYTDKPLMIMPKTGRVHTSYHQAVTATRLSSDTPNLQNIPIVNEEGRRIRQAFIAPEDYIVVSADYSQIELERIMAHLSROKGLLTAFAGKD
IHRATAAEVFLPLFVTVTSEQRSAKAINFGLIYMSAFLGARQINIPRKAQKYMDLYFERYPGVLEYMRTRAQAKQGYVETLDGRRLYLPDIKSSNGARAAAEAAINAPMQGTAAOIKRAMIAVDAMLQAEQPRVRMIMQVHDELVEFHKKDDVDAVAKQIHLNENCTRLDVPLL
VEVSGGENWQAH
>REF_2
MKTUNIDIEYSDEDLTKVGVYKADSPFEILLFAYSVDGQVPECEDLTISEIDPDEIVAAITDKNVLKIAFNAQFVRLSKYLVPPYLDPAQKMTMWHANELGLPASLGQCAKYLNIQEQKDTRGTLINFFSKPCPKTKXNQRTNRLPEHAPEKWTTFIEYCIQDWNEMAJANKJN
RFPVPESEKMLYTLDRQINDRGAEDHETATAIDIMADLSEAGLNEMKELTGLBNPSLAQLKXMLEEQGTTPFEKLQEVVLKALALQNLPENVAEVLKRLSLNSSTKXLYMMDNARCSDNRHIGLQFYGANRTGRWAGALLQVNLPRNYLSEIDFARQLVKAQVEGELIEMVEDPVO
TLKQLIRTLVAKGHRFIVSDFAIEARVIAWYAKQDNLVFRTHGKIYEATAQMFHLGEYTDYDMKSHGDMRQRGVATLALYOGGPGALKAMGALENGIEEHELQDIDVRWRTANKRINFWHTQKAVIDCLQNGGKXGPGRLKPKYKAGFLIQLPSGRKLAYAKHLEKGO
YGPATFYEGGQGVAFTEQTYGKLVENIQATARDLAEAMQRLEREGYIVFHHVDEAVAEVPEGEKSEIEPNEIMSVVPWAEGPLNAEGFETKYMKD
>REF_3
MNIDIEYSSNDISKCGAYKTEADFEILLIAYSIDGGAISAIDMTKVDNEPFFHADFEFKIALFDPVKKYAFNANFERTCLAKHFNKMPPEEWICTMNSMRIGLPASLDKVGVEVLRQKQKAGNLIIRYFSPCKPTKNGGRTNRLPEHDLKXWQFIDYCIROVEVEMTIAHKI
KDFPVTAEQAYWFDQINDRGIKLSKJMLGANVLKQSKHEELNQAKHITGLBNPSPTQLLAWKDDQGLDIPMLQKTVQEYLKEATGKAKKMLEIRLQMSKTSVKYKYNMHMMCSDERVRLGFQFYGAGTGRWAGRGVQLQNLTKHYISDTEIARDLIKEQRFDDLLLNHVP
QDLLSQLVRTFTTAEENELAVSDFSAIEARVIAWYAKQDNLVFRTHGKIYEASQMFNVPVSEITKGDLRQKGVSELALGYOGGAGALKAMGALENGIEEHELQGLVDSMRNANPNIWFMKACQEAADINTVKSARKTHHTGLRFYMKKGLMIELPSGRALAYPKASVGENSGSQ
YVFEFGLDUNRKSGLKTYGKLVENIQATARDLAISIAELEASGFKIVGHVHDEVIEIPRGSNGLKEIETIMNKPVDWAGLNLNSDGFSPFYMKD
>REF_4
MNIDIEYSSNDISKCGAYKTEADFEILLIAYSIDGGAISAIDMTKVDNEPFFHADFEFKIALFDPVKKYAFNANFERTCLAKHFNKMPPEEWICTMNSMRIGLPASLDKVGVEVLRQKQKAGNLIIRYFSPCKPTKNGGRTNRLPEHDLKXWQFIDYCIROVEVEMTIAHKI
KDFPVTAEQAYWFDQINDRGIKLSKJMLGANVLKQSKHEELNQAKHITGLBNPSPTQLLAWKDDQGLDIPMLQKTVQEYLKEATGKAKKMLEIRLQMSKTSVKYKYNMHMMCSDERVRLGFQFYGAGTGRWAGRGVQLQNLTKHYISDTEIARDLIKEQRFDDLLLNHVP
QDLLSQLVRTFTTAEENELAVSDFSAIEARVIAWYAKQDNLVFRTHGKIYEASQMFNVPVSEITKGDLRQKGVSELALGYOGGAGALKAMGALENGIEEHELQGLVDSMRNANPNIWFMKACQEAADINTVKSARKTHHTGLRFYMKKGLMIELPSGRALAYPKASVGENSGSQ
YVFEFGLDUNRKSGLKTYGKLVENIQATARDLAISIAELEASGFKIVGHVHDEVIEIPRGSNGLKEIETIMNKPVDWAGLNLNSDGFSPFYMKD
>REF_5
MNIDIEYSSNDISKCGAYKTEADFEILLIAYSIDGGAISAIDMTKVDNEPFFHADFEFKIALFDPVKKYAFNANFERTCLAKHFNKMPPEEWICTMNSMRIGLPASLDKVGVEVLRQKQKAGNLIIRYFSPCKPTKNGGRTNRLPEHDLKXWQFIDYCIROVEVEMTIAHKI
KDFPVTAEQAYWFDQINDRGIKLSKJMLGANVLKQSKHEELNQAKHITGLBNPSPTQLLAWKDDQGLDIPMLQKTVQEYLKEATGKAKKMLEIRLQMSKTSVKYKYNMHMMCSDERVRLGFQFYGAGTGRWAGRGVQLQNLTKHYISDTEIARDLIKEQRFDDLLLNHVP
QDLLSQLVRTFTTAEENELAVSDFSAIEARVIAWYAKQDNLVFRTHGKIYEASQMFNVPVSEITKGDLRQKGVSELALGYOGGAGALKAMGALENGIEEHELQGLVDSMRNANPNIWFMKACQEAADINTVKSARKTHHTGLRFYMKKGLMIELPSGRALAYPKALVGENSGSQ
YVFEFGLDUNRKSGLKTYGKLVENIQATARDLAISIAELEASGFKIVGHVHDEVIEIPRGSNGLKEIETIMNKPVDWAGLNLNSDGFSPFYMKD
>1
MIVHYAYSQGLDVRISVNGSGEADFEHMAEIAQISRAQKTNLQMYGMKGKLMNGLDKEAEIEVSYQNKV
PFVKQTYNMOKASARGEIKTLGRHCRFPFFPRKFGGFKYKTEEAIDALGHQNYKRAQTYKALNLIQSSAADQT
KKAMVDLYEEDGIIPIHQVHDELNISVBNKEALSIXKMM
>2
LPLNQRGTINDVLDNESLIPAVRAVLEARGIKTSKAVQALGLVAGDGRIRNCLAYHAGTGRWSGRSFQPNLSRG
IKCDVDQVLAETMAGYQPTDDELSTLRACVVGDDGMLTMDYSQIEVRVLAWLQAGQSVLDAFEAGQDIYNNAAKVF
GENEVSQDQNTVKGKPLILGCGFLGSKITFEIFG
>3
KNPTFLTYGGTYGGLTKLGFSEIEAKAIEDSYHALYKVSQDWRKSVQKQJGTYTTVAFGLRVPTILAKTILQNRAT
PYEASQEARMTNAGHQSQYQMLNRAALELQERLFDKSYVYDILPISHIDAQYFLVRNKAAGAKWLNWLTCEWQON
ONIRHEQVLGGQLSIFPYSWKKEYL
>4
MLNNWLDNLQDGRLHGRVNTGATGRMTHSDPMLAQVAGYSPYKEMRSLFTINGKYLVGADAAQLELRLAHYNN
KDYNTIELNGDVHTANQIAAGLDRDRAKTFIYAFLYGAGDAKIGSIVGGSSDDGRTLKARFLSNTPALAGLRQVDDT
>5
MPOSRRHPVDTIVVEYHKGVDVLHQWADLANIKRKAETYNLIGIMYMGVGLAAQLDISKEEAKNLIQHRNVPFV
KQLASIASQRAEDQGTIRTLGRKCRFHLWEPKTFGNKPRLEAKKEYGNINLKRAFTYKALNLIQSSAAD

```

The results of AP – 1 to AP – 5 have responded with the changes in larger datasets in exactly the same manner as demonstrated with a smaller dataset. The results in this validation study have been demonstrated using smaller datasets because this helps explain the basic logic of each of the scripts within the reformatting section of AutoPhy.

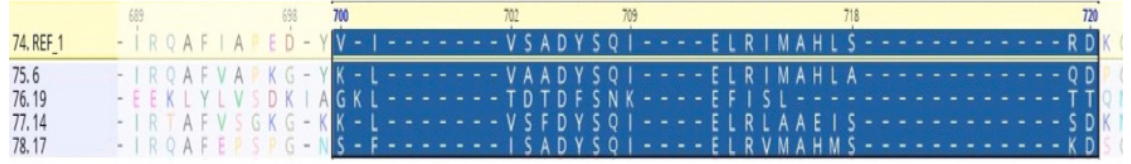

AP – 7: Validating code - autophy-06-trim_reg_int.pl

Function of code: This code has been explained in detail Chapter – 2.

The sequences are chopped to the length of the coverage, specified by the start and end coordinates of the reference sequence.

Expected output: The expected output for this code is a trimmed sequence based on the start and end coordinate.

Table 3.7: Input and output files for AP – 7; output shown for different trimmed regions

Input: multiple sequence alignment in Phylip format (Figure 2.5)	
Output: trimmed sequence; start coordinate = 700 and end coordinate = 720	
REF_1	V-I-----VSADYSQI----ELRIMAHLS-----RD
6	K-L-----VAADYSQI----ELRIMAHLA-----QD
19	GKL-----TDTDFS NK----EFISL-----TT
14	K-L-----VSFDYSQI----ELRLAAEIS-----SD
17	S-F-----ISADYSQI----ELRVMAHMS-----KD
Alignment on geneious	
	
Output: trimmed sequence; start coordinate = 604 and end coordinate = 618	
REF_1	GGAPST-SEEVLEELA
6	KGQPST-AEAVLAELA
19	---HD IENKEV LMM--
14	SGTYST-DSSTLNNLA
17	GGQPST-DEKVLAE LA
Alignment on geneious	
	
Output: trimmed sequence; start coordinate = 85 and end coordinate = 130	
REF_1	PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT----LA-----REAEKA-----GR
6	PDDLRLQIEPLHASVKALGLPLLCDGV--EAD-----DVIGT----LA-----RQSAAS-----GC
19	-----DVELTYQLF-EIF-----LKVFPKKELKVID-----MTLRMF-----ID
14	-----MPIYKEVTIPMEYGVLDLMELLQETHDNIVKDLEENKDVVMKSLLATSEAKW-----VMNTAF-----DNFPPNHKG
17	--MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSPMARPLHKIEV-----GGSLKA-----LA
Alignment on geneious	

		85	95	105	115	116	118	123	128	130				
74. REF_1	KSHRPP	PDDLRAQIEPLHAMVKAMGLPLLAVSGV	--EAD	-----	DV	IGT	----	LA	-----	REA	EKA	-----	GR	
75.6	KSHRPP	PDDLRLQIEPLHASVKALGLPLLCVDGV	--EAD	-----	DV	IGT	----	LA	-----	RQ	SAAS	-----	GC	
76.19			DVELTYQLF	-EIF	-----	LK	VFPKKELKVID	-----	MT	LRMF	-----	ID		
77.14			MP	IYKEVTIPMEEYGV	LDLMDL	QETHDNIVKDL	EENKDVVMKSL	LATSEAKW	-----	VM	N	TAF	-----	DNFPPNHKG
78.17			MDYSDKAILCHNTAF	DGAILSWHFGIKPKLW	-----	LD	TL	SMARPLHKIEV	-----	GG	SLKA	-----	LA	

Output: trimmed sequence; start coordinate = 66 and end coordinate = 110

```

REF_1  GKTFRDELFEDYKSHRPPMPDDLRLQIEPLHASVKALGLPLLCVD
6      GKTFRDELFEDYKSHRPPMPDDLRLQIEPLHASVKALGLPLLCVD
19      -----DVELTY
14      -----MPIYKEVTIPMEEYGVLDLMD
17      -----MDYSDKAILCHNTAFDGAILSWHFG

```

Alignment on geneious

		59	66	75	85	95	105	110
74. REF_1		-----	AVVFDAN	GKTFRDELF	EHYKSHRPPMPDDLRAQIEPLHAMVKAMGLPLLAVSG			
75.6		-----	AVVFDAN	GKTFRDELF	EDYKSHRPPMPDDLRLQIEPLHASVKALGLPLLCVDG			
76.19					DVELTYQ			
77.14					MP	IYKEVTIPMEEYGV	LDLMDL	
78.17					MDYSDKAILCHNTAF	DGAILSWHFG		

Output: trimmed sequence; start coordinate = 695 and end coordinate = 795

```

REF_1  APED-YV-I-----VSADYSQI-----ELRIMAHLS-----RDKGLL--TAFAE--GKDIHRATA-----AE-V-----FGL-P-----
LETVTSE-----QRRSAKAINF-----GLIY--GMSAFGLARQL-----
---NIPRKEAQYMDLYFERY
6      APKG-YK-L-----VAADYSQI-----ELRIMAHLA-----QDPGLL--HAFQN--GLDVHKATA-----AE-V-----FGV-E-----
LDEVSND-----QRRKAKAINF-----GLIY--GMSAFGLAKQI-----
---DVDRKQAQAYIDRYFTRY
19     VSDKIAGKL-----TDTDFSNK-----EFISL-----TTQNIF--KIYPKFSDNSIVAYDC-----LVFI-----HDPNK-----
RPDNLNLCKHVEQANLLTKSSIEEYLNFLSLNYKKIVSEIKLFQSD-----
-----KDLMKLYVELD
14     SGKG-KK-L-----VSFDYSQI-----ELRLAAEIS-----SDKNFI--KAFKN--NEDIHASTA-----KE-I-----FNLND-----
SQINND-----YRRKAKAINF-----GILY--GISPYGLAKQL-----
---DISNTEAKDYINEYL---
17     PSPG-NS-F-----ISADYSQI-----ELRVMAMHS-----KDSGLL--KAFQO--GEDVHSKTA-----SE-V-----FNV-E-----
LDEVTPD-----LRRNAKAINF-----GLIY--GISAFGLGKQL-----
---GISRNLAAYMALYFEKY

```

Alignment on geneious [Alignment was too long to be shown in this section]

695	698	701		785	788	795																	
I	A	P	E	D	-	Y	V	-	I		A	Q	K	Y	M	D	L	Y	F	E	R	Y	
V	A	P	K	G	-	Y	K	-	L		V	A	Q	A	Y	I	D	R	Y	F	T	R	Y
L	V	S	D	K	I	A	G	K	L		L	V	S	D	K	I	A	G	K	L			
V	S	G	K	G	-	K	K	-	L		V	S	G	K	G	-	K	K	-	L			
E	P	S	P	G	-	N	S	-	F		E	P	S	P	G	-	N	S	-	F			

AP – 8: Validating code - autophy-07-ratio_calc_for_selseqs.pl

Function of code: This code is able to calculate the ratio of complete coverage excluding the external gaps for each of the sequences and lists it out in a file separated

by tabs. A regular expression is used to calculate the coverage excluding the external gaps.

To recap through the ratio calculation:

$$\text{Ratio} = \frac{(\text{Aligned amino acids} + \text{internal gaps})}{(\text{End coordinate} + 1) - \text{Start coordinate}}$$

Aligned amino acids + internal gaps, are captured by the following regular expression: `/(\w.*\w)/`

Expected output: Each sequence should have the sequence name, the trimmed sequence, and the ratio calculated associated with it, which uses the regular expression above.

Table 3.8: Input and output files for AP – 8

Input: Trimmed sequence from AP – 8; start coordinate = 700 and end coordinate = 720	
REF_1	V-I-----VSADYSQI----ELRIMAHLS-----RD 1.000
6	K-L-----VAADYSQI----ELRIMAHLA-----QD 1.000
19	GKL-----TDTDFSNK----EFISL-----TT 1.000
14	K-L-----VSFDYSQI----ELRLAAEIS-----SD 1.000
17	S-F-----ISADYSQI----ELRVMAMHS-----KD 1.000
Output:	
REF_1	V-I-----VSADYSQI----ELRIMAHLS-----RD 1.000
6	K-L-----VAADYSQI----ELRIMAHLA-----QD 1.000
19	GKL-----TDTDFSNK----EFISL-----TT 1.000
14	K-L-----VSFDYSQI----ELRLAAEIS-----SD 1.000
17	S-F-----ISADYSQI----ELRVMAMHS-----KD 1.000
Input: Trimmed sequence from AP – 8; start coordinate = 604 and end coordinate = 618	
REF_1	GGAPST-SEEVLEELA
6	KGQPST-AEAVLAELA
19	---HDLENKEVLMM--
14	SGTYST-DSSTLNNLA
17	GGQPST-DEKVLAELE
Output:	
REF_1	GGAPST-SEEVLEELA 1.000
6	KGQPST-AEAVLAELA 1.000
19	---HDLENKEVLMM-- 0.688
14	SGTYST-DSSTLNNLA 1.000
17	GGQPST-DEKVLAELE 1.000

Input: Trimmed sequence from AP – 8; start coordinate = 85 and end coordinate = 130	
REF_1	PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----REA EKA-----GR 1.000
6	PDDLRLQIEPLHASVKALGLPLLVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC 1.000
14	-----MPIYKEVTIPMEEYGVLDLMELLQETHDNIVKDLEENKDVVMKSLLATSEAKKW-----VMNTAF-----DNFPPNHKG 0.939
19	-----DVELTYQLF-EIF-----LKVFPKKELKVID-----MTLRMF-----ID 0.798
17	--MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSMARPLHKIEV-----GSSLKA-----LA 0.980
Output:	
REF_1	PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----
REA EKA	-----GR 1.000
6	PDDLRLQIEPLHASVKALGLPLLVDGV--EAD-----DVIGT-----LA-----
RQSAAS	-----GC 1.000
14	-----MPIYKEVTIPMEEYGVLDLMELLQETHDNIVKDLEENKDVVMKSLLATSEAKKW-----
VMNTAF	-----DNFPPNHKG 0.939
19	-----DVELTYQLF-EIF-----LKVFPKKELKVID-----
MTLRMF	-----ID 0.798
17	--MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSMARPLHKIEV-----
GSSLKA	-----LA 0.980
Input: Trimmed sequence from AP – 8; start coordinate = 66 and end coordinate = 110	
REF_1	GKTRDELFEFYKSHRPPMPDDLRLQIEPLHASVKALGLPLLVD 1.000
6	GKTRDELFEFYKSHRPPMPDDLRLQIEPLHASVKALGLPLLVD 1.000
19	-----DVELTY 0.133
14	-----MPIYKEVTIPMEEYGVLDLM 0.444
17	-----MDYSDKAILCHNTAFDGAILSWHF 0.533
Output:	
REF_1	GKTRDELFEFYKSHRPPMPDDLRLQIEPLHASVKALGLPLLVD 1.000
6	GKTRDELFEFYKSHRPPMPDDLRLQIEPLHASVKALGLPLLVD 1.000
19	-----DVELTY 0.133
14	-----MPIYKEVTIPMEEYGVLDLM 0.444
17	-----MDYSDKAILCHNTAFDGAILSWHF 0.533
Input: Trimmed sequence from AP – 8; start coordinate = 695 and end coordinate = 795	
REF_1	APED-YV-I-----VSADYSQI---ELRIMAHLS-----RDKGLL--TAF AE--GKDIHRATA-----
AE-V	-----FGL-P-----LETVTSE-----QRRSAKAINF---GLIY--GMSAFGLARQL-----
	-----NIPRKEAQKYMDLYFERY 1.000
6	APKG-YK-L-----VAADYSQI---ELRIMAHLA-----QDPGLL--HAFQN--GLDVHKATA-----
AE-V	-----FGV-E-----LDEVSNL-----QRRKAKAINF---GLIY--GMSAFGLAKQI-----
	-----DVDRKQAQAYIDRYFTRY 1.000
19	VSDKIAGKL-----TDTDFS NK---EFISL-----TTQNIF--KIYPKFS DNSIVAYDC-----
LVFI	-----HDPNK-----RPDNLNLCKHVEQANLLTKKSSIEEYLNFLSLNYKKIVSEIKLFQSD-----
	-----KDLMKLYVELD 1.000
14	SGKG-KK-L-----VSFDYSQI---ELRLAAEIS-----SDKNFI--KAFKN--NEDIHASTA-----
KE-I	-----FNLND-----SQINND-----YRRKAKAINF---GILY--GISPYGLAKQL-----
	-----DISNTEAKDYINEYL--- 0.988
17	PSPG-NS-F-----ISADYSQI---ELRVMAHMS-----KDSGLL--KAFQQ--GEDVHSKTA-----
SE-V	-----FNV-E-----LDEVTPD-----LRRNAKAINF---GLIY--GISAFGLGKQL-----
	-----GISRNLA EYMYALFEKY 1.000
Output:	
REF_1	APED-YV-I-----VSADYSQI---ELRIMAHLS-----RDKGLL--TAF AE--GKDIHRATA-----
AE-V	-----FGL-P-----LETVTSE-----QRRSAKAINF---GLIY--GMSAFGLARQL-----
	-----NIPRKEAQKYMDLYFERY 1.000
6	APKG-YK-L-----VAADYSQI---ELRIMAHLA-----QDPGLL--HAFQN--GLDVHKATA-----
AE-V	-----FGV-E-----LDEVSNL-----QRRKAKAINF---GLIY--GMSAFGLAKQI-----
	-----DVDRKQAQAYIDRYFTRY 1.000
19	VSDKIAGKL-----TDTDFS NK---EFISL-----TTQNIF--KIYPKFS DNSIVAYDC-----
LVFI	-----HDPNK-----RPDNLNLCKHVEQANLLTKKSSIEEYLNFLSLNYKKIVSEIKLFQSD-----
	-----KDLMKLYVELD 1.000
14	SGKG-KK-L-----VSFDYSQI---ELRLAAEIS-----SDKNFI--KAFKN--NEDIHASTA-----
KE-I	-----FNLND-----SQINND-----YRRKAKAINF---GILY--GISPYGLAKQL-----
	-----DISNTEAKDYINEYL--- 0.988

```

17  PSPG-NS-F-----ISADYSQI---ELRVMAHMS-----KDSGLL--KAFQQ--GEDVHSKTA-----
SE-V-----FNV-E-----LDEVTPD-----LRRNAKAINF---GLIY--GISAFGLGKQL-----
-----GISRNLAAEYMALYFEKY 1.000

```

AP – 9: Validating code - autophy-08-sel_seqs.pl

Function of code: This code is able to select the sequences based on the threshold mentioned listed out for each of the sequence. A regular expression is used to capture the ratio.

It looks like the following; \$res =~ /(d+|w+) (.*) ([\d.]+)/

Specific bracketed region of interest in regex: BR – 3

Expected output: The digits captured within the BR – 3 have to extract the ratios from output of AP – 8

Table 3.9: Input and output files for AP – 9

Input: ratio calculation output from AP – 9; start coordinate = 85 and end coordinate = 130

```

REF_1  PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----
-----REAEKA-----GR 1.000
6      PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----
-----RQSAAS-----GC 1.000
14     -----MPIYKEVTIPMEEYGVDLDMELLQETHDNIVKDLEENKDVVMKSLLATSEAKKW-----
-----VMNTAF-----DNFPPNHKG 0.939
19     -----DVELTYQLF-EIF-----LKVFPKKELKVID-----
-----MTLRMF-----ID 0.798
17     --MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSMARPLHKIEV---
-----GSSLKA-----LA 0.980

```

This table was taken as an example test run for this script from the AP – 7 and AP – 8. The script has a range of ratios to choose from and different thresholds will be tested to test the durability of the script

Table 3.10: Range of threshold tests

Threshold ≥ 0.95
Expected output: 3 of the sequences from the table above should be chosen; REF_1, 6, and 17
<p>Actual output:</p> <pre> [deepamp@biomix section-2]\$ perl autophy-08-sel_seqs.pl 0.95 >REF_1 PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT----LA-----REAEKA-----GR >17 --MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSMARPLHKIEV-----GGSLKA-----LA >6 PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC </pre>
Threshold ≥ 0.93
Expected output: 4 of the sequences from the table above should be chosen; REF_1, 6, 17, and 14
<p>Actual output:</p> <pre> [deepamp@biomix section-2]\$ perl autophy-08-sel_seqs.pl 0.93 >REF_1 PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT----LA-----REAEKA-----GR >14 -----MPIYKEVTIPMEEYGVLDMEQLQETHDNIVKDLEENKDVMKSLATSEAKKW-----VMNTAF-----DNFPPNHKG >17 --MDYSDKAILCHNTAFDGAILSWHFGIKPKLW-----LDTLSMARPLHKIEV-----GGSLKA-----LA >6 PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC [deepamp@biomix section-2]\$ █ </pre>
Threshold ≥ 1.000
Expected output: 2 of the sequences from the table above should be chosen; REF_1 and 6
Actual output:

<pre>[deepamp@biomix section-2]\$ perl autophy-08-sel_seqs.pl 1.000 >REF_1 PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----REAEKA-----GR >6 PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC</pre>
Threshold ≥ 0.799
Expected output: 4 of the sequences from the table above should be chosen; REF_1, 6, 17, and 14
<p>Actual output:</p> <pre>[deepamp@biomix section-2]\$ perl autophy-08-sel_seqs.pl 0.799 >REF_1 PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----REAEKA-----GR >14 -----MPIYKEVTIPMEEYGVLDLMELLQETHDNIVKDLEENKDVVMKSLLATSEAKKW-----VMNTAF-----DNFPPNHKG >17 --MDYSDKAILCHNTAFDGAISWHFGIKPKLW-----LDTLSMARPLHKIEV-----GGSLKA-----LA >6 PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC</pre>
Threshold ≥ 0.78
Expected output: All 5 sequences from the table above should be chosen; REF_1, 6, 17, 19 and 14
<p>Actual output:</p> <pre>[deepamp@biomix section-2]\$ perl autophy-08-sel_seqs.pl 0.78 >REF_1 PDDLRAQIEPLHAMVKAMGLPLLAVSGV--EAD-----DV-IGT-----LA-----REAEKA-----GR >14 -----MPIYKEVTIPMEEYGVLDLMELLQETHDNIVKDLEENKDVVMKSLLATSEAKKW-----VMNTAF-----DNFPPNHKG >17 --MDYSDKAILCHNTAFDGAISWHFGIKPKLW-----LDTLSMARPLHKIEV-----GGSLKA-----LA >19 -----DVELTYQLF-EIF-----LKVFPKKELKVID-----MTLRMF-----ID >6 PDDLRLQIEPLHASVKALGLPLLCVDGV--EAD-----DVIGT-----LA-----RQSAAS-----GC [deepamp@biomix section-2]\$ █</pre>

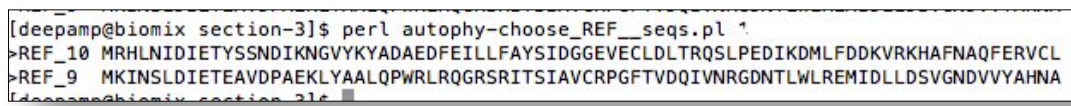
From the results shown above for each of the scripts, it is seen that the trim suite functions in the way expected.

AP – 10, AP – 11 are codes specific to the path followed by AutoPhy for clustering option. The scripts are used to make the flow of data easier through the pipeline [Chapter – 2 for more explanation].

AP – 10: Validating code - autophy-choose_REF__seqs.pl

Function of code: The code specifically pulls out sequences with “REF_” appendage. Basically, this code extracts the reference sequences from the input before multiple sequence alignment is performed.

Table 3.11: Input and output files for AP – 10

Input:
<pre>>87 MILTLDVENTVTERNGKMHLDPFEPDNTLVMVGMLTEAGDETIVTFDHSECAPTDNGRQIVQDMLDQTLLVCHNASH DLVIDTANNLLKLVNKKWMLDFNVPLLEAKIGDNWLDTKSV >88 MILVLDVENTVVERNGKMHLDPFEPENTLVMVGMLDEDGNEDIVTFDHAHKPTLEGRSIVQDKLDGTSLLICHNA AHDVIDMANKNLLKIVNNKWQLDFNVPLLEAKIGPNWLDTKDV >REF_10 MRHLNIDIETYSSNDIKNGVYKYADAEDFEILLFAYSIDGGEVECLDLTRQSLPEDIKDMLFDDK VRKHAFNAQFERVCL >89 MKITLDVENTVTHRDGKMHLDPFVNNSLTMVGMLTDQDDETLVVFDEEAAAPADQESFDLVQSYLDEATVL IMHNAHDGFTSPFYMKD >REF_9 MKINSLDIETEAVDPAEKLYAALQPWRLRQGRSRITSIAVCRPGFTVDQIVNRGDNTLWLREMIIDLLDSVGND VVYAHNA</pre>
Expected output: Sequences with header names starting with “REF_” should get specifically selected
<p>Actual output:</p>  <pre>[deepamp@biomix section-3]\$ perl autophy-choose_REF__seqs.pl 1 >REF_10 MRHLNIDIETYSSNDIKNGVYKYADAEDFEILLFAYSIDGGEVECLDLTRQSLPEDIKDMLFDDKVRKHAFNAQFERVCL >REF_9 MKINSLDIETEAVDPAEKLYAALQPWRLRQGRSRITSIAVCRPGFTVDQIVNRGDNTLWLREMIIDLLDSVGNDVVYAHNA</pre>

AP – 11: Validating code - autophy-13_combine_ref_que.pl

Function of code: This code is used to combine the reference sequences with the query sequences. The reference sequences are added at the beginning, for the purpose of an optimized alignment. Uses AP – 10 output.

AP – 13 is specific to the path of AutoPhy for the non – clustering path.

AP – 13: Validating code - autophy-09-format_fa-phy1.pl

Function of code: This code changes the format of the file from fasta to phylip format.

Table 3.12: Input and output files for AP – 13

Input:	
<pre> >87 MILTLDVENTVTERNGKMHLDPFEPDNTLVMVGMLTEAGDETIIVTFDHSECAPTDNGRQIVQDMLDQTTLLVCHNASHDLVIDTAN NNLLKLVNKKWMLDFNVPLLLLEAKIGDNWLDTKSV >88 MILVLDVENTVVERNGKMHLDPFEPENTLVMVGMLDEDGNEDIVTFDHAHKPTLEGRSIVQDKLDGTSLLICHNAAHDLVIDMAN KNLLKIVNNKWQLDFNVPLLLLEAKIGPNWLDTKDV >REF_10 MRHLNIDIETYSSNDIKNGVYKYADAEDFEILLFAYSIDGGEVECLDLTRQSLPEDIKDMLFDDKVRKHAFNAQFERVCL >89 MKITLDVENTVTHRDGKMHLDPFEVNNSLTMVGMLTDQDDETLVVFDEEAAPADQESFDLVQSYLDEATVLIHMNAHDGFTSPF YMKD >REF_9 MKINSLDIETEAVDPAEKLYAALQPWRLRQGRSRITSIAVCRPGFTVDQIVNRGDNTLWLREMI DLLDSVGNDVVYAHNA </pre>	
Expected output: convert fasta to phylip format	
Actual output:	
<pre> 5 121 87 MILTLDVENT VTERNGKMHL DPFEPDNTLV MVGMLTEAGD ETIVTFDHSE CAPTDNGRQI 88 MILVLDVENT VVERNGKMHL DPFEPENTLV MVGMLDEDGN EDIVTFDHA HKPTLEGRSI REF_10 MRHLNIDIET YSSNDIKNGV YKYADAEDFE ILLFAYSIDG GEVECLDLTR QSLPEDIKDM 89 MKITLDVENT VTHRDGKMHL DPFEVNNSLT MVGMLTDQDD ETLVVFDEE AAPADQESFD REF_9 MKINSLDIET EAVDPAEKLY AALQPWRLRQ GRSRITSIAV CRPGFTVDQI VNRGDNTLWL VQDMLDQTTL LVCHNASHDL VIDTANNLL KLVNKKWMLD FNVPLLLLEAK IGDNWLDTKS VQDKLDGTSL LICHNAAHDL VIDMANKNLL KIVNNKWQLD FNVPLLLLEAK IGPNWLDTKD LFDDKVRKHA FNAQFERVCL ----- LVQSYLDEAT VLIHMNAHD GFTSPFYMKD ----- REMIDLLDSV GNDVVYAHNA ----- </pre>	

AP–L–1: Validating code – autophy-log-2_sel_seqs_less.pl

Function of code: This checks for the number of sequences within the file and if the number of sequences within the file is less than 5, the file does not continue with

phylogenetic analysis. This decision code is a fail-safe option in the pipeline. The code makes sure that enough sequences go into RAxML for phylogenetic analysis.

Table 3.13: logic of AP-L-1

If sequences file > 5
File moves forward for phylogenetic analysis
If sequences in file < 5; the following message is displayed and the pipeline stops
<pre> Pipeline stopped! To keep in mind!: 1.Default threshold level for the selection of sequences is at 0.95. Are you sure you want it to be that strict? 2.The coverage of the sequences, based on the reference sequences is low,and hence the number of sequences to be forwarded to RAxML, is insufficient. </pre>

This code is common between the clustering following path and the path without that option.

AP – 12: Validating code – autophy-10-header_returner.pl

Function of code: This code puts the original header names back into the newick file that is produced as output by phylogenetic analysis.

Table 3.14: Input and Output of AP – 12

Input:
<pre> (((((((82:0.02779055439974751837,20:0.12897776024335469436):0.05626358621016776251, 30:0.19069499680863016833):3.68620792937130303812,77:0.73590479035098710359):0.2322349 6118279995937,(REF_1:0.09734675025793361469,(48:0.25426024525950130517,(100:0.23490393 865680717078,1:0.21485943221708672657):0.32746153214701667622):0.10824733339340221472) :0.25000132124422480562):0.02647819925374549771,(69:0.72024593570220707406,((78:0.0755 1671659185861529,97:0.23728685856228237672):0.14206476806398768420,56:0.00000100000050 002909):2.87896335493411070772):4.13292624736433555910):0.0; </pre>

Output:

```
((((((((CBJ_sngl100000004302_842_3_1:0.02779055439974751837,CBB_sngl100000000709_1_6
69_1:0.12897776024335469436)0.05626358621016776251,CBB_sngl100000001668_1_699_1:0.1906
9499680863016833):3.68620792937130303812,REFERENCE_Escheria_coli_IAI39_DNA_polymerase
I[EscheriacoliIAI39]:0.73590479035098710359):0.23223496118279995937,(CBB_sngl100000005
014_2_709_1:0.09734675025793361469,(CBB_sngl100000004883_86_796_1:0.254260245259501305
17,(CBJ_sngl1000000008238_886_2_1:0.23490393865680717078,CBB_deg7180000000601_1100_2101
_3:0.21485943221708672657):0.32746153214701667622):0.10824733339340221472):0.250001321
24422480562):0.02647819925374549771,(CBJ_sngl100000002029_2_979_1:0.720245935702207074
06,(CBJ_sngl100000003168_1054_2_1:0.07551671659185861529,CBJ_sngl1000000008092_1057_44
_1:0.23728685856228237672):0.14206476806398768420,CBJ_ctg7180000001310_1_1674_1:0.0000
0100000050002909):2.87896335493411070772):4.13292624736433555910):0.0;
```

3.2. Part – 2: Exploring AutoPhy results

Part – 2 is an analysis of the output of AutoPhy. Reliable phylogenetic trees clade in a manner in which evolutionary relationships between sequences can be found. An important point to keep in mind when analyzing phylogenetic trees is the clading of sequences, and if these are grouping in a manner, that tell a story, it is good proof that the sequences in the phylogenetic tree are evolutionarily related and are generally clading according to it. This chapter will be used to analyze the story.

AutoPhy result is analyzed in correlation with data of a manually curated tree in this chapter, in addition to display of the AutoPhy result. Manually curated trees are considered gold standard trees, in phylogenetics. The data when used to make trees manually is continually improved and a manually intensive iterative process of deletion of sequences is used to obtain the optimum tree.

There is always a possibility of variation in the data that has been processed in two different ways. In all processes automatic, there is a probability of missing data, or not considering what the naked eye would. Spontaneity of choice is another important factor to be taken into consideration, in cases where manually curated trees are being created. AutoPhy has been built with the attempt of keeping all of these factors in mind, and more.

This chapter consists of four parts; each part deals with different parts of AutoPhy output and the relevant information from the manually curated tree. Section 3.2.1 discusses background information on the manually curated tree. Section 3.2.2 consists information on the methodology of both the protocols. Section 3.2.3 shows

the results obtained. Section 3.2.4 discusses the results of both the trees and compares them.

3.2.1. Background

DNA polymerase family A is an important family of enzymes that are responsible for synthesizing DNA molecules. In this section, a short introduction of the manually curated tree is given.

Nasko et al (in preparation) have worked on a dataset consisting of both Smithsonian Environmental Research Center (SERC) and MgOl metagenomic sequences. The open reading frames of the datasets were predicted using MetaGeneMark (pap). The sequences were confirmed to a Pol I using NCBI's Conserved Domain BLAST online tool (pap). In this project, a tree was constructed using both the datasets to better understand the global diversity of Pol I.

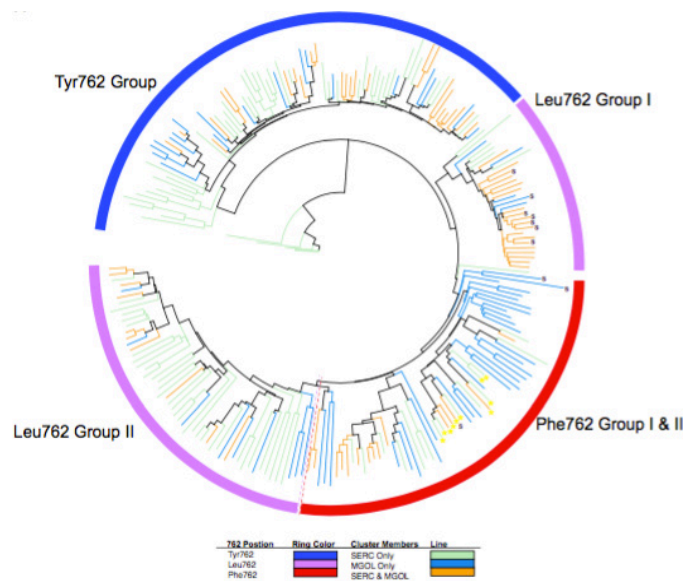


Figure 3.5: Nasko et al (in preparation) Distribution of sequences (same dataset) within the manually curated tree

Figure 3.5, is the manually curated tree developed as a result of this study. The different colored branches in the tree indicate to the source of the sequences (SERC or MgOl). The colored bands around different clades of the tree represent the amino acid present at the 762 position.

Iroki has been used to color the sequences based on the criteria of study/interest in the manually curated tree as well as the AutoPhy output. This tool has been used separately to color the sequences based on the header information provided for the result section. The tool was built with the idea of recognizing a group of sequences with ease. The number of sequences that need to be studied keep increasing and it is easier to make evolutionary inferences when the sequences can be colored based on their label, branch etc., in an automated fashion. Iroki is one such tool [27].

The sequences that have been used in the dataset for the manually curated tree (and AutoPhy) have a unique annotation system. In this system, the 762 residue of the sequences has been mentioned in the header. For e.g. EC_tools_L_utg7180001084019_1_672__435__0__F22 indicates to a “_L_” or “_L” (depending on source of the sequence; SERC or MgOl), which indicates to the specific amino acid in the 762 position. This header annotation system has been used for all amino acids (F, L, Y).

While analyzing global diversity of DNA polymerase family A and other than building a manually curated tree for selected DNA polymerase sequences, factors that were taken into consideration as a part of the study, were the replication systems of the bacteriophage populations. The focus was also on the neighbors of polA and the overall genetic composition of polA containing contigs. It was found that various phage populations had diverse replisome compositions, and this feature could have

possibly been shaped by the preference in hosts and lifestyle by the phage. The phylogenetic relationships among the viruses were investigated.

3.2.2. Methodology exploration

For the manually curated tree a region of 125-aa region was chosen (N675-L799) in the *E coli* Pol I gene product. The sequences were then aligned using MAFFT and clustered at 75% using MOTHUR.

The parameters used in the pipeline (AutoPhy) are the same as that used in the manually curated tree. The region used for trimming was the 125-aa region: N675-L799 in the *E coli* Pol I gene product. Clustering parameter used was 75% and the threshold level of selection of sequences was a strict 100% or 1.000 (depending on coverage percentage or ratio) for complete coverage in terms of length of the region of interest in the reference sequence. The sequences used as reference in the manually curated tree, were also used for AutoPhy validation study.

Table 3.15: Table listing the tools used in both protocols

Tools	Manually curated tree	AutoPhy
Multiple sequence alignment	MAFFT [16]	MUSCLE
Clustering (@75%)	MOTHUR	USEARCH

3.2.3. Results

The sequences were submitted to AutoPhy, with instructions of a strict threshold of 1.000 and 75% clustering, adhering to the protocol followed for the manually curated tree.

Table 3.16: AutoPhy statistics

Statistics at various junctures of AutoPhy	
Input sequences	3260
Reference sequences	11
Output from trim suite at a threshold level of a 100% (1.000 ratio) coverage of region of interest	2517
Clustered output (including reference sequences)	725

Table 3.17: Manually curated tree statistics

Statistics at various junctures of manually curated tree	
Input sequences	3260
Reference sequences	11
Trimming	Strict Iterative process used to choose the sequences of interest
Clustered output (including reference sequences)	181(final result)

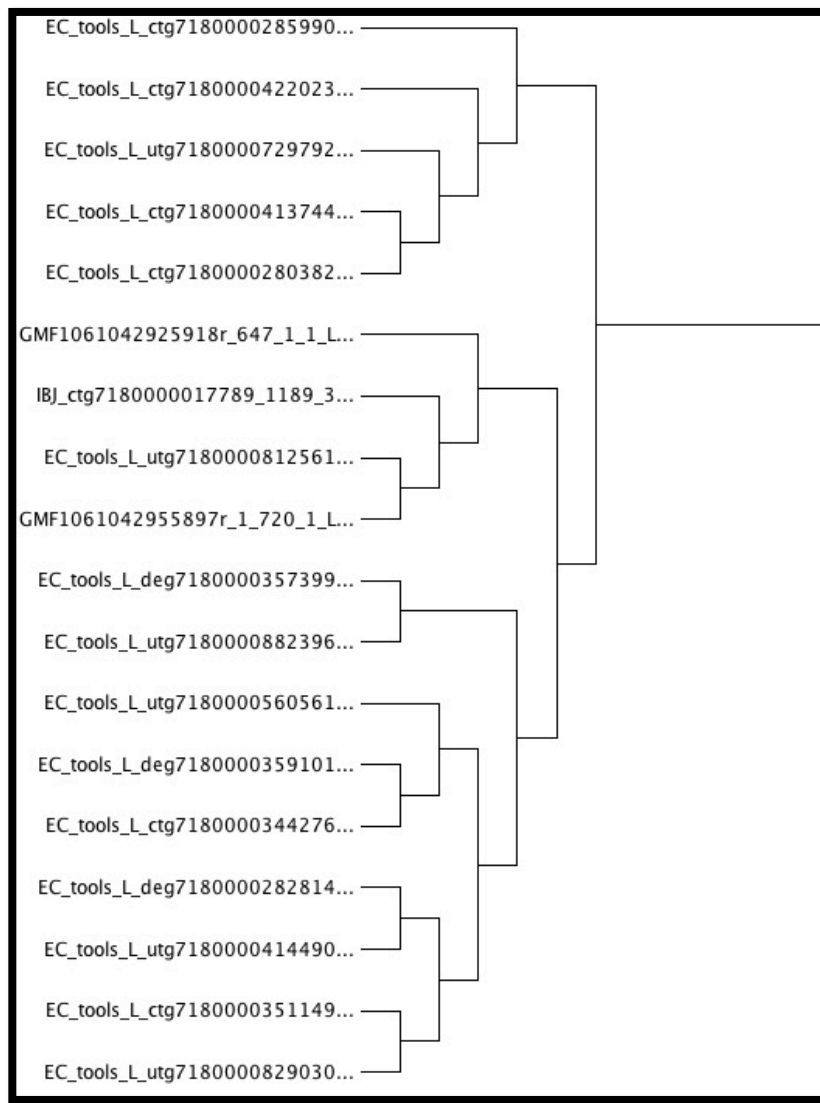


Figure 3.6: Snapshot of a clade, consisting of leucines within the AutoPhy result

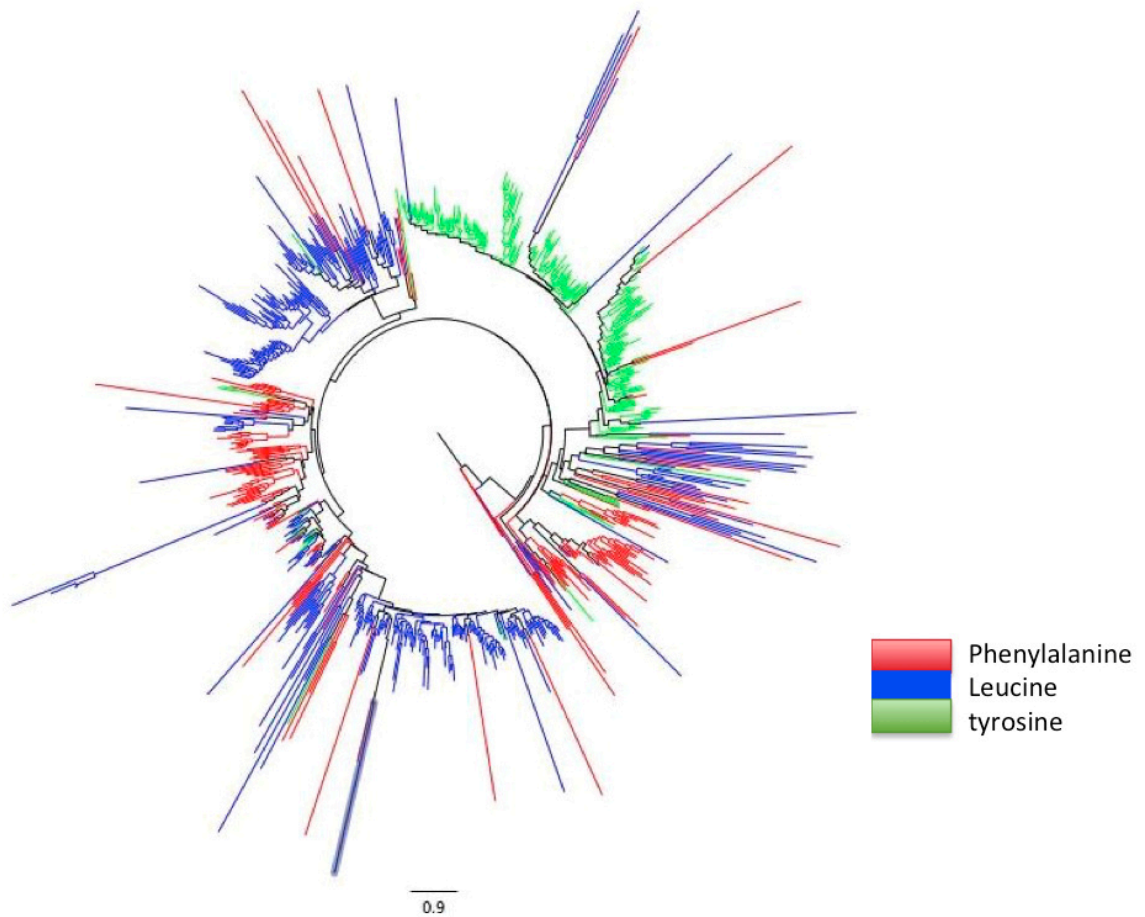


Figure 3.7: phylogram of AutoPhy result (with 725 sequences) including reference sequences

The clade shown in Figure 3.6 is from the phylogenetic tree that consisted 725 sequences (AutoPhy result). Majority of the sequences within the different clades were seen to have uniformity of sequences when amino acid at position 762 was considered.

Figure 3.7 shows the phylogram result output of AutoPhy. There is uniformity in representation of headers in corroboration specific amino acid at the position (explained in section 3.2.1). The trees with different colors depending on the amino

acid mentioned in the header have been made using Iroki. Each branch is colored depending on the amino acid that the sequence header has been assigned. Sequence headers with “_T”(tyrosine) for example have been colored with green.

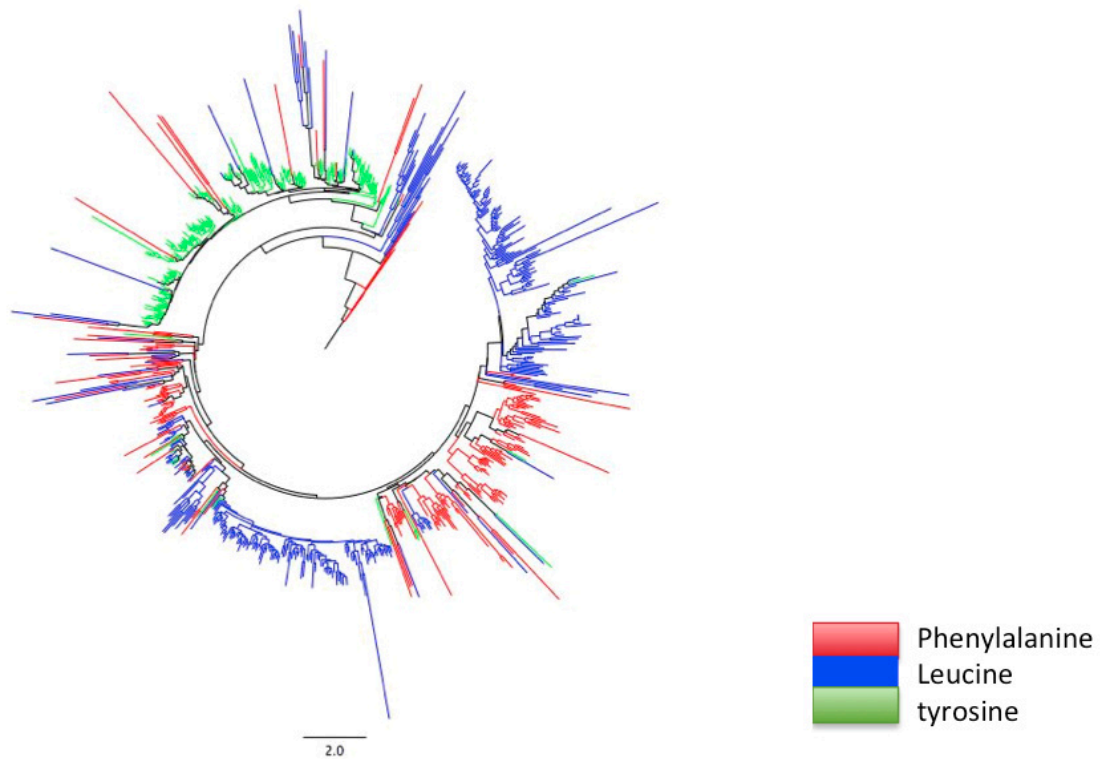


Figure 3.8: phylogram of phylogenetic tree without singletons (with 300 sequences) including reference sequences

The phylogenetic tree shown in Figure 3.8 has been constructed without taking singletons into consideration. This tree was created to view the result without extra

noise created by the deep branches. The total number of singleton sequences was 425. This phylogram has been discussed further in the next section.

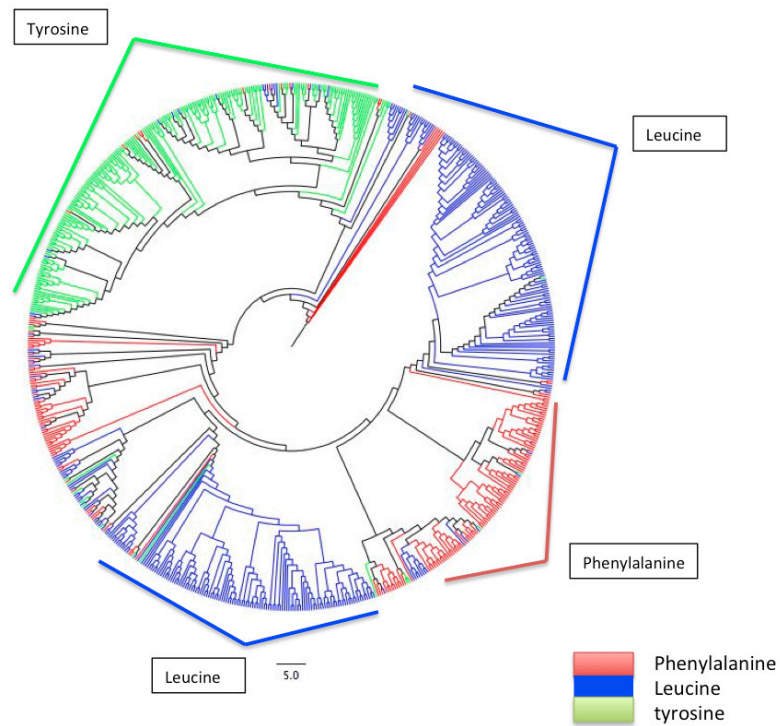


Figure 3.9: cladogram of phylogenetic tree without singletons

The cladogram in Figure 3.9 has been shown to understand the general sequence clading uniformity in Figure 3.8. Discussed further in section 3.2.4.

3.2.4. Discussion

AutoPhy was developed with the aim to capture the marker diversity of interest in shotgun metagenomic data, submitted to VIROME. If the sequences are able to clade in a manner in which information on the potential relationship can be found and if the region of interest in the sequences is widely observed, this could help identify the specific population under investigation.

In this section, the output of AutoPhy and the manually curated tree are analyzed. Various factors involved in both studies have been discussed. This section also discusses the inference from the various trees shown in section 3.2.3 and the issues seen with them currently, and possible solutions.

Figure 3.10 shows all the trees of importance in the study. The figure on the extreme right is the manually curated tree with two leucine clusters and two (exactly next to each other) phenylalanine clusters and one tyrosine cluster. Each of the colored branches represents the source of the sequences. The tree went through an intensive process of manual curation that will be described further in this section.

Figure 3.10 compares the tree without singletons to the grouping pattern of the manually curated tree. Figure 3.8 and 3.9 were developed with the idea of looking at trees without extra noise created by the deeper branches. This tree was built to see the pattern of grouping of the sequences without the noise. While building trees without singletons, the loss of diversity of the dataset was considered, but singleton sequences also tend towards being erroneous, though it might not always be the case. The tree with 725 sequences (Figure 3.7) is seen to have a lot of deep branching and some of the groups have mixed sequences within the clades, even though the general grouping of sequences within clades are observed to group based on the amino acid of interest at

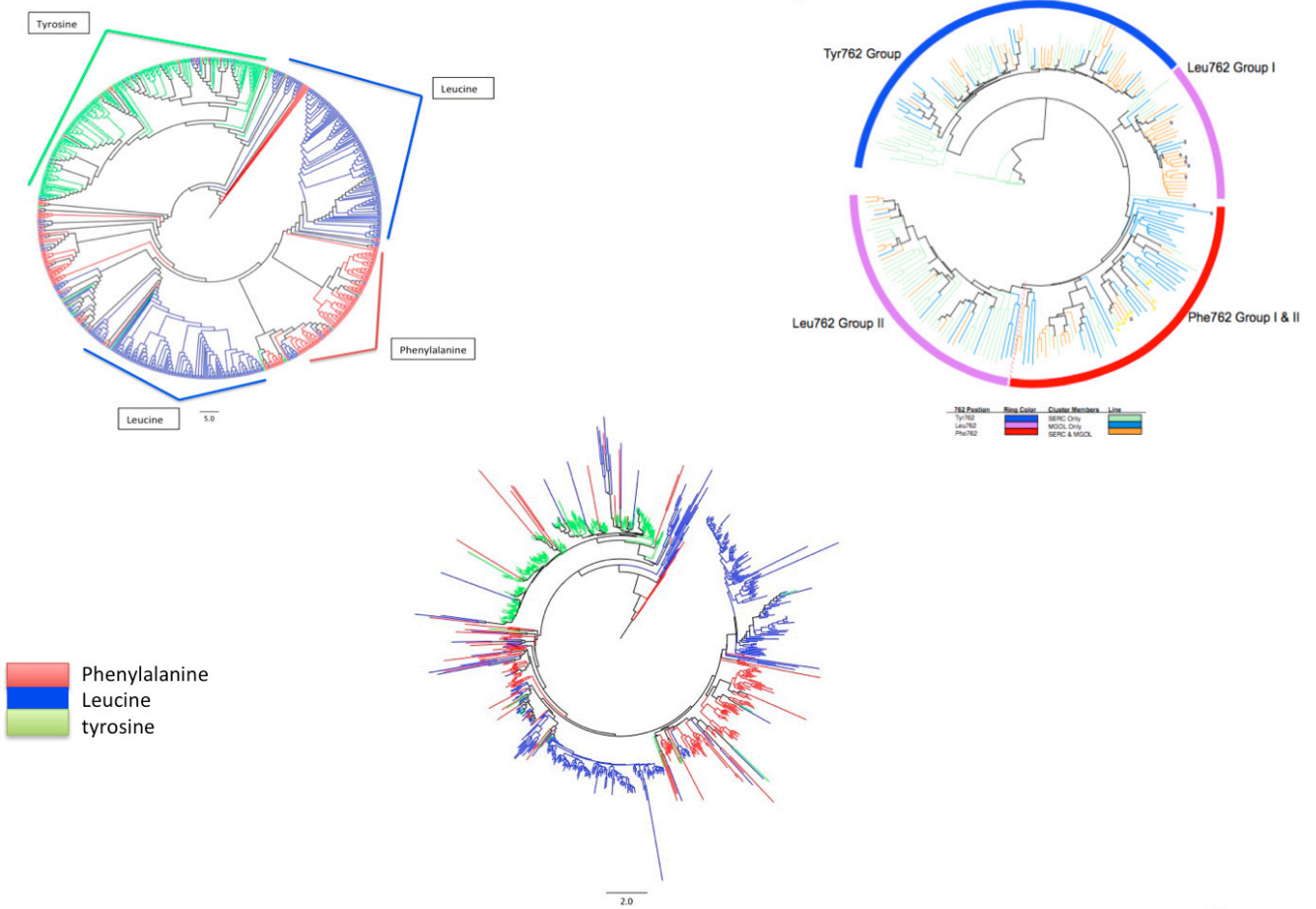


Figure 3.10: This figure shows the three trees of importance in the study and a comparison. The cladogram on the extreme left shows the distribution of sequences according to the amino acid at position 762; the figure on the extreme right shows the distribution of sequences in the phylogenetic tree in the based on the amino acid and source of the sequence

position 762(F, L, Y). The two phenylalanine clades appear separately in the AutoPhy result instead of together as seen in the manually curated tree. The deep

branching and the mixed clades could be occurring because of certain issues. Misalignment could be a possible issue, and the other issue could be the misannotation of the sequences, which means that the sequences might not actually be DNA polymerase sequences. If a different amino acid is present at the position of interest in the alignment, the alignment could workout differently. Misalignment adds deep branches to the phylogenetic tree. The cladogram in Figure 3.10 shows the general grouping of the sequences based on the amino acid of interest.

The main reason contributing to a lower number of sequences (Table 3.18) in the final manually curated tree was that an intensive iterative process of manually deleting sequences occurred. Most deletions were made for the following reasons:

1. Alignment shifted with each iteration
2. Invalid amino acid/gap present at that position

In AutoPhy, before the sequences are submitted to RAxML (right before step AP – 12), multiple sequence alignment is performed on centroid sequences in fasta format. This alignment was viewed on Geneious (Figure 3.11).

359. 2233	-	-	-	T	V	N	F	G
360. 2185	-	-	-	V	V	N	F	G
361. 2444	-	-	-	I	V	N	F	G
362. 2471	-	-	-	T	I	N	F	G
363. 2092	-	-	-	T	I	N	F	G
364. 2141	-	-	-	A	I	N	F	G
365. 2595	-	-	-	A	I	N	F	G
366. 2532	-	-	-	A	I	N	F	G
367. 2429	-	-	-	A	I	N	F	G
368. 2091	-	-	-	A	I	N	F	G
369. 2861	-	-	-	A	I	N	F	G
370. 3137	-	-	-	V	I	N	F	G
371. 3109	-	-	-	A	I	N	F	G
372. 2254	-	-	-	A	I	N	F	G
373. 3248	-	-	-	A	I	N	F	G
374. 3249	-	-	-	A	I	N	F	G
375. REF_1	-	-	-	A	I	N	F	G
376. 2272	-	-	-	A	V	N	F	G
377. 2826	-	-	-	T	F	R	Y	A
378. 2431	-	-	-	-	-	-	-	-

Figure 3.11: snapshot of alignment in Geneious, blue column shows amino acids in the position 762 in different sequences

If we take position 762 as a conformation of alignment of sequences, most of the residues present at position 762 in the alignment (Figure 3.11) were of significance (F, Y, L). The observation made was that a few of the sequences had gaps or different amino acids present at this position.

Table 3.18: snapshot alignment statistics

Total number of sequences	Gap or a different amino acid at position
725	55

This error (Table 3.19) in alignment (and at this position) can be expected from an automated tool, as an automated tool might overlook many issues that the human eye can see. Manually curated trees are always of better standard because of proofing taking place at every step.

If the erroneous sequences were to be removed and multiple sequence alignment repeated, the possibility that the two reasons used for deletion of sequences for the manually curated tree, could stand true for the AutoPhy result as well. Removing these sequences will potentially help with better alignments, when the sequences are submitted for multiple sequence alignment, again. If this process of selection and deletion is performed repeatedly, the sequences selected for the tree ultimately will be highly streamlined. The shift of alignment was one of the main reasons for deletion of sequences, and this can be expected when the process is repeated after the removal of faulty alignments.

Chapter 4

CONCLUSION

AutoPhy was developed with the aim to capture the marker diversity of interest in shotgun metagenomic data. To elucidate this process a phylogenetic tree was created. The hope was that this phylogenetic tree would be able to show the evolutionary relationship within the clades and sequences. If the sequences are able to clade in a manner in which information on the potential relationship could be found, the marker is worth taking into consideration. If the region of interest in the sequences is widely observed, this could be a region in the sequences that could help identify the specific population under investigation.

AutoPhy would be run as a middle layer in the VIROME pipeline and not as a standalone tool, with strict upstream analysis of sequences before getting submitted to the tool.

As a part of validation the perl scripts that were integrated into the pipeline produced output according to the expected output. The trim suite specifically, was able to select sequences based on the threshold level given with precision (Chapter -3, Part -1:Validation of scripts).

The output of AutoPhy was also examined in correlation with the data of the manually curated tree.

There are a few points to be kept in mind before concluding this chapter.

A. Aim of AutoPhy

The preliminary aim was to build a tool that would be able to quantify the region of interest (in marker gene). The tool should have been able to survey the dataset submitted and to be able to recognize the region of importance based on the reference sequence. It should also be able to produce a phylogenetic tree. AutoPhy was able to perform the above functions (Figure 3.5 and 3.6).

B. Curating the tree

Automatic tools are not able to cover all the conditions based on which trees when manually curated are covered. They are not able to include spontaneity of decision and the keen observation power of the human eye when looking at very specific point.

In the Discussion section of the previous chapter a few important things were taken into consideration.

- 762 position
- Reasons for deletion of sequences from manually curated tree

When the multiple sequence alignment was viewed on Genious to survey the amino acids at the 762 positions of all the sequences, based on the *E coli* reference, it was observed that the majority of the sequences contained the amino acid of interest at the position (F, L, Y). It was also observed that there were sequences, which were not annotated with the correct amino acid or had a gap at the position (Figure 3.11). Majority of the sequences having the significant amino acid at the position was able to

prove that AutoPhy was working as expected for a “first cut” and the erroneous placement at this position the fallibility of automatic tools and need for improvement.

During the development of the manually curated tree the sequences in the dataset had undergone repeated multiple sequence alignment, clustering and trimming with a very strict criterion for deletion of sequences. The sequences were deleted whenever necessary and this included shifting of alignment or the absence of an amino acid of significance from the 762 positions.

Considering all of these points, suggestions for future improvements of AutoPhy to achieve cleaner and better standard trees have been listed:

1. Adding a script to the pipeline to remove gap or alternate amino acid in specific position of interest, if necessary
2. Trim suite and multiple sequence alignment executed iteratively
3. Testing with a different aligner for multiple sequence alignment
4. Using a different clustering algorithm

The difference in results could have been brought about by the difference in the aligner used, difference in the clustering algorithm used or the difference in the number of times trimming and multiple sequence alignment performed. The aim of comparing the result of an automated tool and manually curated tree is to observe if the automated tool is group sequences within the clades in a similar manner. From the result section the conclusion is that AutoPhy is able to do the necessary.

The fact of deep branches and mixed clades cannot be ignored, and for the first version of the tool, AutoPhy is able to perform reasonably. To better the performance of the tool, in terms of refining the number of sequences and deep branching, the future improvements suggested would be a place to begin.

The multiple sequence alignment tool and the clustering algorithm used in the pipeline was based on the decision that the tool should be able to handle large datasets. Both USEARCH and MUSCLE are known to be better with large datasets [17].

Another point to be kept in mind before concluding is the position specificity that could be considered as a possible step in the tool. In the validation study, it was seen that performance/accuracy of the tool was being measured by position 762. If a step is added for position specificity then this condition could be further satisfied. This step would involve reading the alignment right after the clustering step. If the position specified as an important residue required for the description of the sequence functionally, is to be taken into consideration, then the sequence with the expected amino acid at that position only, would be selected. If the sequence does not contain the expected amino acid, then the sequence would be rejected. The selected sequences would then undergo clustering once again, before an alignment step. This could be repeated until the position of interest consists of the sequences with only the amino acid of importance.

The development of AutoPhy was an important step towards automation of phylogenetic tree construction and identifying regions of interest (marker genes) within metagenomic sequences. The methodology chapter in this thesis gives an in depth look into the logic and result of each of the steps of the pipeline. The validation was divided into two parts. The first part dealt with the individual validation of each of the scripts in the pipeline. The second part involved exploring different parts of the AutoPhy output.

The conclusion is that AutoPhy is able to produce a functional phylogenetic tree that is capable of telling a story.

REFERENCES

1. Ackermann HW. Bacteriophage observations and evolution. *Res Microbiol.* 2003;154(4):245-251. doi:10.1016/S0923-2508(03)00067-6.
2. Breitbart M. Marine Viruses: Truth or Dare. *Ann Rev Mar Sci.* 2012;4(1):425-448. doi:10.1146/annurev-marine-120709-142805.
3. Bexfield N, Kellam P. Metagenomics and the molecular identification of novel viruses. *Vet J.* 2011;190(2):191-198. doi:10.1016/j.tvjl.2010.10.014.
4. Schoenfeld T, Liles M, Wommack KE, Polson SW, Godiska R, Mead D. Functional viral metagenomics and the next generation of molecular tools. *Trends Microbiol.* 2010;18(1):20-29. doi:10.1016/j.tim.2009.10.001.
5. Polson SW, Wilhelm SW, Wommack KE. Unraveling the viral tapestry (from inside the capsid out). *ISME J.* 2011;5(2):165-168. doi:10.1038/ismej.2010.81.
6. Labonté JM, Reid KE, Suttle CA. Phylogenetic analysis indicates evolutionary diversity and environmental segregation of marine podovirus DNA polymerase gene sequences. *Appl Environ Microbiol.* 2009;75(11):3634-3640. doi:10.1128/AEM.02317-08.
7. Ruperao P, Edwards D. Bioinformatics: Identification of Markers from Next-Generation Sequence Data. In: *Methods in Molecular Biology (Clifton, N.J.).* Vol 1245. ; 2015:29-47. doi:10.1007/978-1-4939-1966-6_3.
8. Sakowski EG, Munsell E V, Hyatt M, et al. Ribonucleotide reductases reveal novel viral diversity and predict biological and ecological features of unknown marine viruses. *Proc Natl Acad Sci U S A.* 2014;111(44):15786-15791. doi:10.1073/pnas.1401322111.
9. Soltis DE, Soltis PS. The role of phylogenetics in comparative genetics. *Plant Physiol.* 2003;132(4):1790-1800. doi:10.1104/pp.103.022509.
10. Dutilh BE, Snel B, Ettema TJG, Huynen MA. Signature genes as a phylogenomic tool. *Mol Biol Evol.* 2008;25(8):1659-1667. doi:10.1093/molbev/msn115.

11. Schmidt HF, Sakowski EG, Williamson SJ, Polson SW, Wommack KE. Shotgun metagenomics indicates novel family A DNA polymerases predominate within marine viroplankton. *ISME J.* 2014;8(1):103-114. doi:10.1038/ismej.2013.124.
12. Roux S, Faubladier M, Mahul A, et al. Metavir: a web server dedicated to virome analysis. *Bioinforma Appl NOTE.* 2011;27(21):3074-3075. doi:10.1093/bioinformatics/btr519.
13. Wommack KE, Bhavsar J, Polson SW, et al. VIROME: a standard operating procedure for analysis of viral metagenome sequences. *Stand Genomic Sci.* 2012;6(3):427-439. doi:10.4056/sigs.2945050.
14. Edgar RC. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* 2004;32(5):1792-1797. doi:10.1093/nar/gkh340.
15. Edgar RC. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics.* 2010;26(19):2460-2461. doi:10.1093/bioinformatics/btq461.
16. Katoh K, Misawa K, Kuma K, Miyata T. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.* 2002;30(14):3059-3066. <http://www.ncbi.nlm.nih.gov/pubmed/12136088>. Accessed April 3, 2017.
17. Thompson JD, Linard B, Lecompte O, Poch O. A comprehensive benchmark study of multiple sequence alignment methods: Current challenges and future perspectives. *PLoS One.* 2011;6(3). doi:10.1371/journal.pone.0018093.
18. Balcazar JL (2014) Bacteriophages as Vehicles for Antibiotic Resistance Genes in the Environment. *PLOS Pathogens* 10(7): e1004219. doi: 10.1371/journal.ppat.1004219
19. Ajith Harish, Aare Abroi, Julian Gough, Charles Kurland; Did Viruses Evolve As a Distinct Supergroup from Common Ancestors of Cells?. *Genome Biol Evol* 2016; 8 (8): 2474-2481. doi: 10.1093/gbe/evw175
20. National Research Council (US) Committee on Metagenomics: Challenges and Functional Applications. *The New Science of Metagenomics: Revealing the Secrets of Our Microbial Planet.* Washington (DC): National Academies Press (US); 2007. 1, Why Metagenomics? Available from: <https://www.ncbi.nlm.nih.gov/books/NBK54011/>
21. Winter C, Bouvier T, Weinbauer MG, Thingstad TF. Trade-offs between competition and defense specialists among unicellular planktonic organisms: the “killing the winner” hypothesis revisited. *Microbiol Mol Biol Rev.* 2010;74(1):42-57. doi:10.1128/MMBR.00034-09.

22. Roux S, Faubladier M, Mahul A, et al. Metavir: a web server dedicated to virome analysis. *Bioinforma Appl NOTE*. 2011;27(21):3074-3075. doi:10.1093/bioinformatics/btr519.
23. Zhong, X., & Jacquet, S. (2013). Prevalence of Viral Photosynthetic and Capsid Protein Genes from Cyanophages in Two Large and Deep Perialpine Lakes. *Applied and Environmental Microbiology*, 79(23), 7169–7178. <http://doi.org/10.1128/AEM.01914-13>
24. Yoshida-Takashima Y, Yoshida M, Ogata H, Nagasaki K, Hiroishi S, Yoshida T. Cyanophage infection in the bloom-forming cyanobacteria *Microcystis aeruginosa* in surface freshwater. *Microbes Environ*. 2012;27(4):350-355. doi:10.1264/jsme2.me12037.
25. Li S. Viruses of other microorganisms.
26. Punta M, Coggill PC, Eberhardt RY, et al. The Pfam protein families database. *Nucleic Acids Res*. 2012;40(D1):D290-D301. doi:10.1093/nar/gkr1065.
27. Moore, R. M, A. O. Harrison, S. M. McAllister, R. L. Marine, C. Chan, and K. E. Wommack. 2017. Iroki: automatic customization for phylogenetic trees. *bioRxiv* 106138, doi:10.1101/106138
28. Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B., Wu, C. H., & the UniProt Consortium. (2015). UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6), 926–932. <http://doi.org/10.1093/bioinformatics/btu739>
29. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*. 1990;215(3):403-410. doi:10.1016/S0022-2836(05)80360-2.