HIV CLASSIFICATION USING DNA SEQUENCES

by

VERONIKA NOVOSELSKY

(Under the Direction of Liang Liu)

ABSTRACT

Many phylogenetic methods used for HIV classification analyze a collection of whole genome sequences to classify a new virus. Since the computational speed is reduced when we analyze whole genome, methods that analyze only some genomic regions were developed. Phylogenetic analysis based on complete genome is more reliable than those based on short segments of the HIV genome.

We propose a new phylogenetic classification method based on coalescent theory. We choose the best-fitted model for every gene segment of the sequences using AIC criterion. Then we use maximum likelihood estimation to infer a phylogenetic tree and K-means clustering to classify the sequences. We observed significant improvement in HIV string classification by utilizing information provided by the entire genome. We take advantage of the whole genome and also recognize the uniqueness of every gene region. We tested the method on 150 sequences sampled from Los Alamos HIV database and obtained 100% subtyping accuracy.

INDEX WORDS: Coalescent, Phylogenetic analysis, Distance matrix, HIV classification

HIV CLASSIFICATION USING DNA SEQUENCES

by

VERONIKA NOVOSELSKY

BS, Pace University, 2003

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2013

HIV CLASSIFICATION USING DNA SEQUENCES

by

VERONIKA NOVOSELSKY

Major Professor:	Liang Liu
Committee:	Paul Schliekelman Pengsheng Ji

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
August 2013

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION

The etiologic agent of AIDS was first discovered in 1983-1984. Since then, Human Immunodeficiency virus, HIV, has affected millions of people worldwide and posed significant public health challenges (Schochetman, 1994). In 2010, there were 872,990 persons in the United States diagnosed with HIV infection. Among these, 47,500 were diagnosed in 2010. The number of deaths of persons ever receiving a diagnosis of AIDS was 15,529 in 2010 alone (CDC, 2012). Worldwide, for the year 2011, approximately 34 million people were living with HIV infection and only about 50% of them knew of their HIV status. In the same year, there were estimated 2.5 million new HIV infections and the number of Aids-related deaths was 1.7 million (UNAIDS, 2013). As of today, despite much research, HIV vaccine trials have failed to provide full protection.

The genes of the virus are encoded in the RNA genome. The HIV-1 genome structure is composed of 9 genes (Schochetman, 1992; Bushman 2012) as seen in Figure 1.1:

GAG - Core protein

POL - Enzymes

ENV - Envelope proteins

TAT - Positive regulator

REV - Differential regulator

VIF - Infectivity factor

VPR - Not Known

VPU - Not Known

NEF - Negative regulator

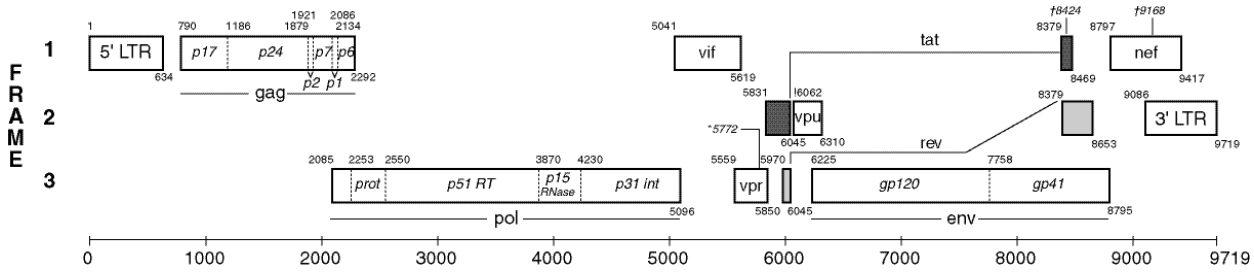


Figure1.1: Schematic representation of HIV-1 genome structure

The defining feature of HIV-1 infection is its exceptional diversity during the course of infection of a single individual. The remarkable genetic diversity stems from at least four sources: high substitution rate (0.002 substitutions/sites/year), a rather small genome (9.8 kilobase in length), short generation times (0.1-0.2 mutations/genome/generation) and high recombination frequency (Castro-Nallar, 2012). When HIV was discovered, it had the fastest rate of evolution in any eukaryotic cell. For example, the scale of HIV-1 diversity that accumulates in one infected individual exceeds that of all influenza virus isolated worldwide in any given year (Korber, 2001). The overall single step point mutation rate for HIV-1 is ~$3x10^{-5}$ mutation per base per replication cycle (Mansky, 1995). That translates to about one genome in three containing a mutation after one replication cycle. Frame shifts, duplications and inversions are also common form of mutation in HIV viruses (Svarovskaia, 2003). Another feature of HIV is the ease of generating large number of sequences, about 10^{11} viruses are produced daily. In

addition to that, accumulation of diversity varies between regions, with rates being much higher for *env* region and lower for *gag* and *pol* regions. In addition, *env* is characterized by structural changes such as insertion, deletion and duplication which tend to increase its length (Bushman, 2012). These characteristics complicate HIV classification and influenced the methods we used in this thesis.

Genetically diverse pathogens like HIV are often stratified into phylogenetic subtypes for classification purposes. Understanding the evolution of HIV and determining its genetic subtype is crucial for reconstructing its origin, interpreting its interaction with the immune system, as well as for epidemiological monitoring and developing effective control strategies and potential vaccines. We are proposing a Phylogenetic method based on Coalescent theory. Maximum Likelihood is used for phylogenetic tree estimation. K-mean clustering is used for classification. We have achieved a perfect classification for a collection of HIV-1 sequences randomly sampled form Los Alamos HIV database.

CHAPTER 2

LITERATURE REVIEW

2.1 Coalescent Theory

The similarities found in a collection of homologous DNA sequences give us information about the evolutionary history of those sequences. We usually can infer which sequences are most closely related to each other. We can also deduct how far back in time the common ancestors of different sequences occurred. If the sequences are from individuals of the same population we get genealogical information, and in this case we can construct gene trees. Understanding the statistical properties of the gene genealogy is the main topic of coalescent theory. Coalescent theory was first introduced in 1980s as an approximation of the Wright-Fisher model for large populations (Kingman, 14). Since 1980, it evolved to incorporate variations in population size, recombination and selection (Neuhauser, 1997).

2.2 HIV Classification Methods

Over the years a number of conceptually and technically different techniques have been proposed for determining the HIV subtype; however, there is not a universally optimal approach and there are no comprehensive comparative benchmarking studies for subtyping methods in the literature. Numerous HIV classification approaches can be theoretically categorized by whether or not they use a phylogeny, whether or not they require a multiple sequence alignment and by

the level of automation. Table 1 categorizes some of the techniques that are widely used for classification purposes.

Table 1: HIV Classification Methods

Non-Phylogeny Methods	*Fully Automated Techniques*	*Phylogeny Methods*
jpHMM	STAR	REGA
STAR	REGA	Selected regions such as *env* or *gag* or *pol*
MuLDAS	BLAST	Whole genome. Concatenation Method
BLAST	jpHMM	Whole genome. Extract richest evolutionary information
	SCUEAL	SCUEAL

2.2.1 Non-Phylogeny Methods

The defining characteristic of Non-Phylogeny methods is the use of training datasets that are composed of well-studied sequences employed to train statistical genotype models that will be used to classify new unknown sequence data. Non-Phylogeny based methods require a large number of reference sequences for accurate prediction (Bushman, 2012).

One of the examples of fully automated Non_Phylogeny method is jpHMM,Classification method. jpHMM is based on profile Hidden Markov Model in which subtype determination is based on the clustering of the unknown sequence with sequences of known subtype. Profile HMM, the training set, is constructed from multiple sequence alignments (MSA) of a set of similar sequences that are approximately the same length, share the same characteristics and can capture the unique features of the set. Later the profile can be used to determine whether an unknown sequence can be considered to be homologous to the set of sequences from MSA.

The procedure, step by step:

1) For each HIV subtype, several known sequences are downloaded from reliable sources, such as Los Alamos HIV database.

2) *gag-pol* regions of these sequences are selected and aligned using MUSCLE or other alignment packages. We will call the group of aligned homologous sequences the MSA.

3) Then a pHMM profile is created for each MSA using HMMER program. The profile is a series of nodes. Each node corresponds to a position (column) in the alignment. There is a transition probability associated with the profile HMM. If probability of transition from one node to another is 1, the path through the model is linear and model starts checking the match for the next node. Furthermore, when we align a new sequence to a profile HMM we take the most probable path that sequence may take through the model. If the new sequence is equivalent to the one in MSA, the model will pass from match node to match node in a linear fashion. If the sequence cannot pass through some node, the probability of that sequence to be similar to the sequences in MSA is lowered. When a sequence cannot pass through all nodes, we would say that it is not homologous to the sequences in MSA.Finally, a score is developed to make a classification decision (Dwivedi, 2012).

Other examples of Non-Phylogeny method are STAR and MuLDAS.STAR uses a *pol* region to subtype sequences relative to a large set of accurately defined reference sequences. Unlike other methods, STAR uses amino acid sequences (Myers, 2005). MuLDAS classifies a query sequence based on the statistical genotype models learned from reference sequences. Its classification model is based on the distances between the query sequence and the known sequences; it employs Bayesian methods (Kim, 2010).

2.2.2 Phylogeny Methods

Literature refers to molecular phylogenic reconstruction as being "the most theoretically sound" approach in determining HIV relatedness (Schochetmen, 1994). Molecular Phylogenetic method provides researchers with powerful tools to analyze if sequences form a monophyletic group. Related HIV viruses would be expected to position the closest to each other, which is visualized via phylogenetic method of tree-building.

The underlying core of numerous published phylogenetic HIV classification methods consists of the pairwise distance method and a parsimony method. In the first method sequences are compared with each other and the degree of their difference is represented as a single number, with large number conveying higher dissimilarity between sequences. Tree-drawing algorithm optimizes the tree topology and branches according to this number. The second method uses raw non- summarized sequence data for building the trees. The method pays attention to sequence patterns assuming affiliation of sequences which share the same pattern (Bushman, 2012).

Phylogenetic approaches are labor intensive, so approximate approaches have been developed to address the issues of automation, speed and complex phylogenetic definition of recombined viruses. Automatic methods utilize modern programming languages such as Java and Pearl for seamless integration of classification algorithm with web interfaces for instant, user friendly output delivery.

There are many phylogenetic methods. Some of them use whole genome sequences, while others utilize partial genome information. Next, two Phylogeny methods, one of which is fully automated, will be covered in more details. The first illustrated method is a manual

phylogenetic method that is based on the whole genome sequence. Even though it is based on the whole genome, the method implies that not all the strings in the genome contribute equally to the evolutionary distance calculations. Some have more discriminatory power than others.

The first step of this method is to extract the strings that identify the richest evolutionary information and then to use only those selected strings in the evolutionary distance calculations.

There are three distinct methods in the literature to define the most important strings. One such method uses relative entropy (or Kullback-Leibler distance) to assign importance to a string in the genome.

After the most important strings in the genome are selected, we calculate their appearance probabilities in the genome. These probabilities are called composition values. They are stored in a sequential order in a vector called the Complete Composition Vector (CCV). At the completion of this step, the whole genome is represented by that vector.

Then, a distance matrix is build using the elements of Complete Composition Vector. Finally, the distance matrix is fed to the Neighbor-Joining algorithm to display the phylogenetic relationships among the whole genome sequences. That concludes the HIV-1 subtype prediction (Wu, 2007).

The last described method, REGA, is an automatic phylogenetic method that is also based on the whole genome. First, the new unidentified sequences are aligned using the popular software Clustal. Then, the alignment of the entire genome sequence is used to build a phylogenetic tree. A single nucleotide substitution model, Hasegawa, Kishino and Yano (HKY) is employed in this step for any input data variations. Repeated clustering is applied to form snug clusters, which are considered “pure” subtype and discrimination rules are applied to identify CRF and unidentified HIV subtypes. Finally, bootstrap and maximum likelihood methods are

used to verify the accuracy of assigned subtypes. REGA software suite runs on powerful machines with UNIX, Linux or Mac operating system. Java based software interface provides a user-friendly access to REGA tool (Oliveira, 2006).

CHAPTER 3

DATA

We collected full gene sequence data from Los Alamos database. Then, nine gene regions were extracted and converted into PHYLIP format. See Figure B.1 in APPENDIX B for the flowchart of the process.

3.1 Database

Over the past years, HIV data has been growing grew exponentially, creating one of the largest databases of epidemiological information. Two main databases have been created to help researchers study large HIV variation. One is HIV RT/Protease Sequence Database in Stanford http://hivdb.stanford.edu, which specializes on sequences associated with the development of resistance against anti-retroviral drugs. The other one is Los Alamos HIV Sequence Database http://www.hiv.lanl.gov, which is customized for data analysis and classification. Majority of sequences in Los Alamos HIV databases are classified by the original authors who undoubtedly used various methods. The Los Alamos database is updated biweekly with Genbank data, and is considered the richest database of HIV data. In 2011, it was reported to hold 414,398 sequences, which was an increase of 22% since 2010 (Kuiken, 2011). Testing our method required a large number of high quality sequences with assigned genotypes; hence, our algorithm was tested with the sequences of HIV-1 from Los Alamos.

In addition to being a database, Los Alamos provides a library of HIV research related articles and tutorials, as well as a large number of phylogenetic tools that are accessible via the

web. Gene Cutter is one of such interfaces available from Los Alamos database and was used in this research. This tool slices coding regions from a nucleotide alignment. To define the start and end of genes, Gene Cutter uses Hmmer v 2.32 as well as Gene coordinates (see Figure1.1) from the HIV reference sequence HXB2. HXB2 is a sequence derived from the first HIV-1 and is a standard reference strain.

3.2 Data Collection

A dataset consisting of 6 batches of 25 sequences was obtained from the Los Alamos database. Each batch has five subtypes, and for each subtype we randomly selected five sequences from the pool of sequences including circulating recombinant forms (CRFs) with the complete genome region. The sequences defined as problematic sequences by the search engine were not return; hence were not considered for the samples. Sequences were downloaded in FASTA format without marked protein regions. See APPENDIX A for a detailed list of the 150 downloaded sequences.

The "Sequence Search interface" page can be found at http://www.hiv.lanl.gov/components/sequence/HIV/search/search.html where users specify optional search requirements. From the genomic region drop down box we selected the option 'complete genome'. We left Virus box as a default, which is HIV-1. Then, we clicked the 'Subtype' box to select a particular Subtype, and pressed CTR to select multiple subtypes. Additional options could be found under the 'Advanced' button. Also, sequences search can be customized by patient, geographic origins, length and other information that is tracked by the database. Here the number of outputted sequences can be changed from the default settings by editing the number in the List box. I used 4000.

After submitting the query, the data were returned in the form of a table of records containing 9 columns of data. The data returned included (i) hyperlinks to the Patient history, Blast and Genomic Region, (ii) Sequence source of origin including Country, Sampling Year (iii) a classification of the sequence by Subtype and Organism, and (iv) additional data depending upon the query. In addition to the tabular results, users have the option of sorting the data by each column. Results can be saved on a local computer in various formats. I selected FASTA format since Gene Cutter, that we used in the next step, takes input data in that format.

At the download time, by default, Los Alamos created multiple sequence alignments of 25 sequences and saved them in one file. The start of each sequence was marked with the custom selected sequence name, which consisted of a random number followed by the sequence subtype. I have selected this naming convention for further formatting reasons.

3.3 Data Preparation

3.3.1 Gene Cutter

Next, we extracted nine genomic regions using the extraction tool, the Gene Cutter. It accepted aligned sequences in FASTA format. Los Alamos returned aligned sequences by default, otherwise we would have used Clustal program in order to shift gaps.

We uploaded the file that was created in section 3.2 using Gene Cutter interface (http://www.hiv.lanl.gov/content/sequence/GENE_CUTTER/cutter.html). Contiguous Sequenced fragments were disassembled into nine regions and stored in 9 separate files, one file

for each HIV-1 gene region. These files along with the original full genome file were zipped and returned to the user via email almost instantly.

3.3.2 Formats

For each of the six samples, the concluding step in preparing the data for classification analysis was converting each of 10 files from FASTA into PHYLIP format. This was done because the software used to analyze the data and build phylogenetic trees takes input files in PHYLIP format. Both FASTA and PHYLIP formats are a text-based format. Some programs on the market take data in one format but not the other.

A sequence in FASTA format begins with a single-line description followed by lines of sequence data. The description line for this project is the custom created sequence name. With FASTA format, the sequence name begins with than (">") separating the name from the sequence data. Examples of sequences in FASTA format are:

>A.CD.1997.97CD_KTB13.AM000054ATGAAAAgGAACTATCAACACTTATtTGTGCCAA
A>A.SN.2001.DDJ369.AY521631GTAAAATG>A.UG.2007.p9004SDM.JX236676ATGAGA
GTGAgGGGGATACAGAgGGgGcCATGGTTTTGGGGATGATAATAATtTGTAGTGcT

The first line of the input file in PHYLIP format contains the number of HIV sequences and the number of characters separated by blanks. Each DNA sequence starts with a ten-character virus name. Examples of sequences in PHYLIP format are:

25 2758
A.CD.1997.ATGAAAGTGAgGGGGATACAGAgGAA

A.SN.2001.ACGAGAGTGAtGGGGATACAGAgGAA

A.UG.2007.ATGAGAGTGAgGGGGATACAGAgGAA

---------GaAcCTTGATTTTTGGGATTATAATAATtTGTAGTGcTGc-

---------GgAtAATATTCCTTGGATTGATAATAATtTGTAAGGcTAc-

---------GgGcCATGGTTTTGGGGATGATAATAATtTGTAGTGcTGc-

On the PC, a format conversion was performed using Phylogeny.fr available as a freeware at http://www.phylogeny.fr/version2_cgi/data_converter.cgi. On UNIX, a format conversion was performed using Clustal using the following command line:

clustalw -infile=DATA_FASTA/VPU.NA.FASTA -type=dna -outfile=VPU.NA.PHYLIP -output=PHYLIP

infile=DATA_FASTA/VPU.NA.FASTA is the full path and the name of the input file

outfile=VPU.NA.PHYLIP is the full path and the name of the output file

This run was done for *vpu* region as reflected in the name of the input and output files.

Other available programs are Format Converter v2.0.5 a tool at Los Alamos and Alignment format converter at http://biotechvana.uv.es/servers/afc/main.php. All these programs take as input the sequences in one format and convert the sequences to a different user-specified format.

CHAPTER 4

METHODS

After preparing the data, we performed model selection for building the tree for each gene in the the batch. Then we created distance matrices capturing node information from the gene trees. Finally, we applied clustering algorithms to classify each gene. See Figure B.2 in APPENDIX B for the flowchart of the process.

4.1 Phylogeny

Molecular phylogeny is the science of estimating evolutionary histories using DNA and amino acid sequences. The earlier applications of phylogeny to HIV classification and origin inference dates back to early 1990s (Huet, 1990). Today phylogenetic analysis has become a common practice of many HIV/AIDS research programs in particular for classification within HIV diversity.

Given a collection of HIV viruses, the objective of a phylogenetic analysis is to produce an evolutionary tree that graphically depicts the genealogical relationships among these viruses within the alignment. A tree is a mathematical structure consisting of internal nodes, external nodes (leaves) and branches, where external nodes are the input HIV sequences. A tree topology which includes the graph and the leaf labels represents all of the ancestral relationships between HIV sequences. A branch connecting two nodes encrypts the amount of change that occurred between the nodes. The branch length represents genetic distance or evolutionary time. Branch lengths and tree structure are calculated from the

alignment and can vary depending on the model is used to build the tree. During a phylogenetic search, numerous candidate trees exist, each representing a hypothesis of the true tree.

4.2 Alignment

Before a tree is build, the sequences must be aligned. (I did not explicitly align sequences as noted in sections 3.2 and 3.3.1.) An accurate alignment is the first step in making a proper and correct analysis of HIV datasets. Many alignment software programs align the input sequences so that the corresponding positions are in same column and pad missing nucleotide as nulls where needed. After sequences are matched, each "column" becomes a single character in the phylogeny computation. If two sequences have a common ancestor they are said to be homologous. By aligning them, we are inferring positional homology from statistically significant sequence similarity; any two sequences have some measurable similarity, but a statement of homology implies that this similarity is a specific result of common ancestry.After an alignment has been generated, the appropriate model of sequence evolution should be selected.

4.3 Model Selection

Modeling has a long history in statistic and also has been used in phylogenetic analysis. In phylogenetic analysis, modeling describes the different probabilities of change from one nucleotide to another, which are caused by evolution, along a phylogenetic tree. Elements such as rate of evolution, branch length, and tree topology are represented by parameters in a model. The phylogenetic model estimates the parameters used to find the best suited tree. A higher rate

of evolutionary change increases the number of parameters used and consequently, increases model complexity. Model selection for a particular dataset is adapted to the features of the data, which is true in statistic in general.

The choice of nucleotide substitution model plays a major role in the results of analysis. Selecting a different model could lead to constructing a different tree which could support a different hypothesis. The decision of how to select among different models should be based on statistically justified criteria such as sequential likelihood ratio tests (LRTs), Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC) and performance-based decision theory (DT). According to publications, (Posada, 2004), AIC has advantages over other selection criteria. BIC criteria, as a rule, selects the model with fewer parameters, which usually are more general models.

We used AIC model selection criteria. AIC quantifies the lost information when a specific model is used to approximate the actual evolution process of the HIV virus. Therefore, among all considered models, we favored the one with the lowest AIC.

Phylogenetic statistical modeling is unique and standard statistical software packages cannot be used. One of the reasons is that the main parameter in phylogeny is the tree topology, which is discrete, while standard software is dominated by continuous parametric models. There are several open access phylogeny statistical modeling tools available on the internet such as Modeltest, MRBAYES, PAUP and MEGA2**.** jModelTest and Phyml were used for this project.

jModelTest is a statistical tool we used for selecting the models of nucleotide substitution. On the backend, jModelTest integrates ReadSeq, PhyMl, and Ted for computational processes. PhyML produces phylogenetic trees based on maximum likelihood from alignments of HIV nucleotide sequences. Leaves will correspond to the HIV viruses, each node represents a

common ancestor, and the branch length is the period of time over which HIV virus has evolved. The main advantage of PhyML is the large number of substitution models to fit the phylogenetic tree typology.

To illustrate model selection, we created the phylogenetic relationships of 25 HIV DNA sequences. The best fitted model for these data was selected using the AIC after calculating likelihood scores of 88 models using jModelTest**.** Finally, the selected model is implemented in PHYLIP.

jModelTest has a PC window interface version, which is suitable for small jobs, and the command line version, which is mostly used on UNIX for bigger jobs. Using SSH Secure File Transfer Client, we FTP to UNIX the 6 previously created batches, each containing 10 files in PHYML format. jModelTest takes files in PHYML format only.

On UNIX, we run jModelTest for each file totaling 60 runs using the following command: *java -jar ~/jmodeltest-2.1.1/jModelTest.jar -d ENV.PHYLIP -s 11 -g 4 -i -f –AIC* This particular command was executed for *env* region as reflected in the name of the input file

This tested 88 gamma models with 4 parameters for every gene region of HIV sequence and selected one based on AIC criteria. Hence the model selection was custom tailored for every gene region of HIV sequence.

The output file for each run contains setting information and model statistics such as AIC value, -lnL value, p value and gamma value. The list of tested models is sorted by AIC criteria with the best proposed model being on top. For example, see Table 2 below for the list of 10 best models selected for Batch 6. We noted the name of the best model selected. That name was used as one of the parameters in the next step, PhyML execution.

Table 2: Batch 6. Best Models based on AIC

Region	*Model*	*-lnL*	*K*	*AIC*	*delta*	*weight*	*cumWeight*
GAG	GTR+I+G	10454.9981	58	21025.9962	0	0.994	0.994
NEF	TIM2+G	5612.7071	55	11335.4142	0	0.3072	0.3072
POL	GTR+I+G	17040.1942	58	34196.3883	0	0.994	0.994
REV	TVM+G	3135.5968	56	6383.1937	0	0.4412	0.4412
TAT	TVM+G	2662.5032	56	5437.0064	0	0.4142	0.4142
VIF	GTR+I+G	3997.1367	58	8110.2734	0	0.5889	0.5889
VPR	TIM1+I+G	2002.4571	56	4116.9142	0	0.2446	0.2446
VPU	GTR+G	2543.2551	57	5200.5101	0	0.6492	0.6492
ENV	TVM+I+G	25982.5776	57	52079.1552	0	0.7311	0.7311

Alignments of HIV DNA sequence file became the input file for PhyML, and those were the same files we used for running jModelTest. The program customized the output based on the user's specified parameters. I had 5 types of the output files for each run : tree file and the corresponding statistic file, as well as the boot trees file, consensus file, and the consensus corresponding statistic file.

On UNIX, we run PhyML for each HIV genome region, totaling 60 runs using the following command: */usr/local/phyml/latest/bin/phyml -i DATA_PHYLIP/POL.NA.PHYLIP -d nt -m 012345 -ae -ve -b0*

This particular command was executed for *pol* region as reflected in the name of the input file using GTR+G model, which had 012345 as its numeric representation. When running bootstrap analysis the -b 0 was replaced with -b100 for 100 bootstrap trees replicates.

4.3.1 Computational approaches to tree estimation. Maximum likelihood

There are a number of computational approaches to phylogenetic tree estimation. Among these are distance-based methods, Neighbor joining, Maximum Parsimony Method and

Maximum Likelihood Methods. Among them, Maximum Parsimony and Maximum Likelihood methods are character state methods for building trees, which means that both methods require discrete characters, DNA sequences.

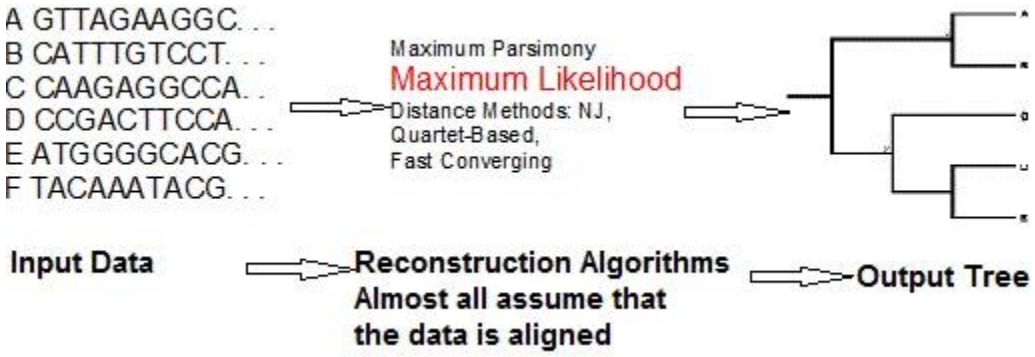


Figure 4.1: Process flow of phylogenetic tree estimation

The maximum likelihood (ML) method of phylogenetic tree construction is not as popular as other tree construction methods because the algorithms are time-consuming which is hard to implement with the large number of sequences under consideration. However, maximum likelihood estimation has several advantages which makes it the method of statistical inference most applicable to HIV data.

One of the main advantages is the efficient use of sequence data. Maximum likelihood makes use of the full data to infer phylogenies; therefore it requires fewer gene sequences than summary statistics methods to achieve a certain level of confidence.

Parsimony methods implicitly assume the amount of change is small over the evolutionary times being considered. On the contrary, HIV data involves large amounts of change, and it is in such cases that parsimony methods can fail. When amounts of evolutionary change is large it can be shown (Felsenstein, 1978) that parsimony methods make an inconsistent estimate of the evolutionary tree, converging to the wrong tree. In contrast, the ML method is

characterized by statistical consistency (Felsenstein, 1981; Hasegawa, 1991; Yang, 1994; Rogers, 1997) and robustness to violations in the assumptions of the underlying evolutionary models (Hasegawa, 1991; Yang, 1994).

Maximum Likelihood Methods use an explicit statistical model based on log likelihood. Maximum likelihood estimation involves finding that evolutionary tree which gives the highest probability of observed DNA sequences. Maximum likelihood , customized for this thesis is the following:

Input: A set of HIV DNA sequences S1, S2, . . . , Sm, which are strings of binary data.

We are looking for a tree typology and branch lengths such that the likelihood, log2L(S1, S2, . . . , Sm | , *tree typology*, branch lengths) is maximized (Felsenstein, 1981).

Many tree topologies are examined. The tree with the highest likelihood score is the tree with the highest probability of producing the input alignment. However the number of possible tree topologies increases rapidly with the number of sequences.

Among available software packages to estimate phylogenetic trees based on the maximum likelihood methods are PAL, Geneious, PAUP, Phylip/dnaML and PAfastdnaML.

Once a tree had been generated the next step was to use the tree viewing software to help view and analyze the tree. Many such software programs are available on the internet. Figtree was used for this thesis.

4.4 Distance Matrix

For further HIV classification, the ML HIV gene trees were used to build a distance matrix, where we assumed that gene trees were known without error. When calculating the distance matrix, the tree topology, which represent the branching order and branch lengths may

be considered. Distance methods that have been proposed in the literature are not limited to Tripartition distance, where successor nodes are divided into strict and non-strict descendants (Moret, 2004); Path-multiplicity distance, which takes into account the number of paths from every node to each leaf and Robinson-Foulds distance that counts the number of branches that occur in one tree are also distance methods.

Branch lengths reflect expected amount of evolution; different branches may have different rates of evolution. When evolutionary rates vary from site to site, which is one of the main characteristic of the HIV data, the tree reconstruction cannot precisely estimate branch lengths of the species tree. For our analysis, we ignored branch lengths because it is difficult to precisely estimate branch lengths of the tree from sequence data. Thus adding branch lengths in the analysis will introduce excessive estimation noise. The theoretical foundations for omitting the branch length can be found in (Liu, 2011.).

For the two concluding steps of classification of the downloaded sequences, we FTP ML tree files from UNIX back to PC and processed them with R.

We had 25 species of HIV viruses. A gene tree was built for each genomic region of HIV virus totaling in nine trees. We assume that all nine regions of the virus are available, which was guaranteed by the sampling method. Let $\{g_k, k = 1,\ldots,9\}$ be the 9 gene trees in the data set. The viruses are denoted by Xj (j= 1,…,25). Leaves in a gene tree represent HIV viruses sampled from Los Alamos database. The distance $D_k(X_{j1}, X_{j2})$ between two leaves *Xj1* and Xj2 in the gene tree k is defined as the number of internal nodes of k between Xj1 and Xj2. Final distance $D_{\text{final}}(Xj1, Xj2)$ between HIV viruses Xj1 and Xj2 is defined as the average of the distances between Xj1 and Xj2 found in the 9 trees corresponding to the genomic regions. In equation (1), $D_k(X_{j1}X_{j2})$ is the internode distance between two HIV viruses X_{j1} and X_{j2} for gene tree *k*.

$$D_{final}(X_{j1}X_{j2}) = \frac{\sum_{k=1}^{9} D_k(X_{j1}X_{j2})}{9} \quad (1)$$

As an example, for the five viruses named n894140 n1591690 n89420 n894130 n894180 sampled from HIV-1 type O, the distance between n894140 and n1591690 is 1 for *env* region, 3 for *pol* region and 2 for *rev* region (See Figure C.1 in APPENDIX C). Similarly, the distance between n894140 and n894130 is 4 for *vpr* region, 3 for *vif* region, and 4 for *tat* region. Also, the corresponding entry for these 2 viruses is lower in *env* matrix ,1, than the matching entry in the *pol* matrix ,3.

The smaller the distance between two HIV viruses, as measured by the number of the internal nodes, the closer they should be to each other in the tree. That also can be interpreted that there has not been much time for the evolutionary change. In the distance matrix it is reflected by the smaller numeric entry. Based on these observation, for HIV strings n894140 and n1591690, there has been less genetic change (mutation, drift, linkage) in *env* region compared to *pol* region.

The average distance between virus n894140 and n1591690 is equal to (1+4+1+3+2+2+2+1+1)/9 = 1.889. Following this procedure, we can calculate all pairwise average distances among all 25 HIV viruses and construct a distance matrix of average distance (See Figure C.2 in APPENDIX C).

4.4.1 Distance Matrix R Implementation

Next, we will cover R implementation of Distance calculation.

1. Phyml-produced tree files were read into R using phybase package commands

R> nodematrix<-read.tree.nodes(tree,taxaname)$nodes

This command produced a hierarchical table, nodematrix. This table has kept track of each node and its ancestor nodes via special coding.

2. R program reads that "special coding" and counted the number of internal nodes between leaves based on the "special coding" rules.

As a result of this step, we have got 9 distance matrices, 1 for each genomic region.

3. In our concluding step, we calculated the Full Genome Distance matrix

R> distTOT<-distENV +distGAG +distNEF +distPOL +distREV +distTAT +distVIF +distVPR +distVPU

distAVG=distTOT/9

Where dist* are the names of distance matrices for each gene region

4.5 Clustering

In our next and final step for HIV strings classification we defined a sequence type by grouping a collection of unlabeled data. We organized HIV sequences in the group based on the distance entries in the distance matrix. In our case, we predefined 5 clusters into which the HIV virus can be divided. Distance is the similarity criterion. Viruses belonging to the same cluster are "close" or "similar" according to a distance.

There are several popular clustering algorithms, for instance Fuzzy C-mean, Hierarchical Clustering and Mixture of Gaussians . We use K-mean clustering because it is easy to explain and implement. In addition to that, the classification made by the K-mean algorithm perfectly matched the Los Alamos classification. The algorithm is fast and is based on the distance matrix. Other clustering algorithms have not demonstrated yet undisputable advantage over the K-mean clustering algorithm. Finally, K-mean is an exclusive clustering algorithm, meaning if one virus belongs to one group (cluster), it cannot belong to another group. This K-mean quality supports our desired classification result.

We selected k=5 central points, where k corresponds to the number of clusters. Then each data point was linked to the nearest center. That completed the initial grouping. The second iteration started by recalculating the k cluster centers, which were taken as being the centroids of the clusters defined in the previous step. Then again we recalculated the distance from each data point to the new centers and redefined the cluster's components based on the shortest distance. We looped through these steps until centroids could not be moved anymore. These steps resulted in defining k homogeneous groups in the collection of unlabeled data.

4.5.1 Clustering R Implementation

Next, we will cover K-mean R implementation:

1. As a result of calculating Full Genome Distance, we have got 6 25-by-25 full genome distance matrices (five HIV viruses for each 5 subtypes, which became an input data into K-clustering program. (See Figure 4.2)

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	...	...	[,23]	[,24]	[,25]
[1,]	0	7	7	6	3	7	...	...	6	7	8
[2,]	7	0	1	2	5	5	...	...	6	7	8
[3,]	7	1	0	2	5	5	...	...	6	7	8
[4,]	6	2	2	0	4	4	...	...	5	6	7
[5,]	3	5	5	4	0	5	...	...	4	5	6
[6,]	7	5	5	4	5	0	...	...	6	7	8
...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...
[23,]	6	6	6	5	4	6	...	...	0	2	3
[24,]	7	7	7	6	5	7	...	...	2	0	2
[25,]	8	8	8	7	6	8	...	...	3	2	0

Figure 4.2: Distance Matrix

We randomly assigned 5 initial cluster-centers out of 25 HIV viruses

R> center<-sample(1:ndata,k)

> center

[1] 11 20 1 15 16

Meaning, our initial cluster centers will be HIV viruses(data points) whose names were ordered under numbers 11, 20 1, 15 and 16.

2. Held the first center fixed and found the virus that was at the furthest distance from virus (data point) marked as 11. (See Figure 4.3)

R> center[i]<-which(newdist==max(newdist))[1]

	...	[,11]	...
[1,]	...	9	...
[2,]	...	9	...
[3,]	...	9	...
[4,]	...	8	...
[5,]	...	7	...
[6,]	...	9	...
[7,]	...	9	...
[8,]	...	8	...
[9,]	...	9	...
[10,]	...	1	...
[11,]	...	0	...
[12,]		11	
[13,]	...	11	...
[14,]	...	9	...
[15,]	...	9	...
[16,]	...	10	...
[17,]	...	10	...
[18,]	...	10	...
[19,]	...	8	...
[20,]	...	8	...
[21,]	...	8	...
[22,]	...	6	...
[23,]	...	4	...
[24,]	...	3	...
[25,]	...	2	...

Figure 4.3: Center Search

As seen from Figure 4.3, the first occurrence of the longest distance from data point 11 and all other data points is 11, which corresponded to the 12th data point. Therefore, our initial cluster center 20 is replaced with 12. (See Figure 4.4)

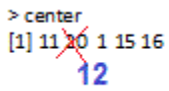


Figure 4.4: Center Replace

3. Find the 3rd data point that was at the biggest distance from the points 11 and 12. To accomplish this, we associated the remaining 23 data points to either point 11 or 12 based on the minimum distance criteria. (See Figure 4.5)

R>newdist<-as.vector(apply(newdist,1,min))

	[,11]	[,12]	
[1,]	9	9	same
[2,]	9	7	
[3,]	9	7	
[4,]	8	6	
[5,]	7	7	
[6,]	9	5	
...	....	....	

Figure 4.5: Center Search Continue

For example, if the distances between data points 1,11 and data points 1, 12 are equal, then the data point 1 belongs to 1st ordered data point which is 11. In the next row the distances between data points 2,11 and data points 2, 12 are 9 and 7. Since 7<9, the 2nd data point belonged to the 12th data point. Then, the 6th data point belonged with the 12th one. Following the same logic, we associated each data point to either center 11 or center 12 based on the shortest distance from each data point to those two centers.

As a result of these interim steps, we obtained an array of minimum distances between each data point and the centers 11 and 12.

[1] "center"

[1] 11 12

[1] 9 7 7 6 7 5 5 8 9 1 -1 -1 1 9 9 4 2 4 8 8 8 6 4 3

Then we selected the first occurrence of the maximum distance, which was 9. It was in the first position of the array; therefore, it was associated with data node 1.

R> Center(i)<-which(newdist==max(newdist))[1]

Therefore, our updated centers were 11, 12 and 1.

4. After redefining all 5 centers as being the furthest from each other, we associated each of the remaining 20 data points with one of those centers.
5. Each remaining data points belonged to the closest center. (See Figure 4.6)

Cluster:	1	2	3	4	5
	[,11]	[,12]	[,1]	[,8]	[,2]
[1,]	9	9	0	8	7
[2,]	9	7	7	8	0
[3,]	9	7	7	8	1
[4,]	8	6	6	7	2
[5,]	7	7	3	6	5
[6,]	9	5	7	8	5
[7,]	9	5	7	8	5
...	...	...	...	...	...
[25,]	2	10	8	7	8

Figure 4.6: Data Points and Cluster Centers

The data point 1 in the first row was the center itself, hence the min distance equals to 0. Therefore, the data point 1 belonged to the cluster 1 with it being the cluster center.

The data point 2 in the second row was also a center itself, hence the minimum distance equals to 0. Therefore, the data point 2 belonged to the cluster 5 with it being the cluster center.

The data point 3 in the third row had the minimum distance equals to 1 which corresponded to the data point 2 in the distance matrix. Therefore, the data point 3 belonged to the cluster 5 .

The data point 7 in the seventh row had the minimum distance equals to 5 which corresponded to the data point 12 in the distance matrix. Therefore, the data point 7 belonged to the cluster 2.

The data point 25 in the twenty fifth row had the minimum distance equals to 2 which corresponded to the data point 11 in the distance matrix. Therefore, the data point 25 belonged to the cluster 1 .

As a result of these steps, we obtained an array of data points. The original number of each data point in the array was hidden and instead was coded as 1,, 2, 3, 4 or 5. The new naming convention for sequences was the cluster number each data point was assigned to at the completion of this step.

R implementation of above:

R > cluster<-apply(newdist,1,order)[1,]

> print(cluster)

*[1] **3 5 5** 5 3 2 **2** 4 3 1 1 2 2 3 3 2 2 2 4 4 4 4 1 1 **1***

As a result, we had 5 clusters with points associated to each cluster.

6. We calculated the distance from each data point to all other points within a single cluster. The point with the smallest distance to all other data points within a cluster became the new center.

For example, one of the clusters consisted of the data points 10, 11, 23, 24, 25 ..

R> "x" x<-which(cluster==i)

[1] 10 11 23 24 25

Distances between each data points in this cluster were the following (See Figure 4.7):

R> [1] "y" y<-dist[x,x]

	10	11	23	24	25
10	0	1	4	3	2
11	1	0	4	3	2
23	4	4	0	2	3
24	3	3	2	0	2
25	2	2	3	2	0

Figure 4.7: Distance within a Cluster

For example, distance between point 10 and all other data points was 1+4+3+2=10 The distance between point 25 and all other data points was 2+ 2+3+2=9

R> z<-apply(y,1,sum)

10 10 13 10 9

Therefore point 25 was the new center, since 9 is the smallest number in the array above.

7. We repeated step 7 for all 5 clusters.
8. We went back to assign each data point to the new centers from steps 7 and 8.f. Then we kept redefining the clusters by repeating step 5 through 8. We looped through steps 5 to 8 until newly defined cluster centers stopped changing.

R> Diff<-summ(ans(oldcenter-center)

Do while diff>0

In conclusion, each of 25 sequences in all six batches were assigned to one of the 5 clusters.

CHAPTER 5

RESULTS

In this section we will show the HIV classification results based on our proposed method, classification results based on the individual gene, and finally classification results based on concatenation method.

5.1 Results Our Proposed Method

The following tables Table 3- 14 show classification results and misclassification rate for each batch. From the tables Table 4, 6, 8, 10, 12, 14 we see perfect classification for all batches.

Table 3: Batch 1

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
n34227533	CRF33_01B	4
n34227633	CRF33_01B	4
n9014933	CRF33_01B	4
n9015133	CRF33_01B	4
n9015233	CRF33_01B	4
n35058735	CRF35_AD	3
n35059035	CRF35_AD	3
n49675335	CRF35_AD	3
n49676035	CRF35_AD	3
n6609735	CRF35_AD	3
n118481A	A	5
n118482A	A	5
n118483A	A	5
n157009A	A	5
n157010A	A	5

n100906F2	F2	1
n149263F2	F2	1
n203821F2	F2	1
n494848F2	F2	1
n494866F2	F2	1
n159169O	O	2
n89413O	O	2
n89414O	O	2
n89418O	O	2
n89424O	O	2

Table 4: Misclassification Summary for Batch 1

		Predicted Subtype				
Real Subtype		*Subtype 1*	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
	Subtype 1	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

Table 5: Batch 2

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
n10258001	CRF01_AE	2
n10622801	CRF01_AE	2
n14200001	CRF01_AE	2
n7766301	CRF01_AE	2
n8943801	CRF01_AE	2
n181594F1	F1	1
n284633F1	F1	1
n288671F1	F1	1
n312761F1	F1	1
n62007F1	F1	1
n101448G	G	4
n50585G	G	4

n88145G	G	4
n88826G	G	4
n89440G	G	4
n14889N	N	3
n159509N	N	3
n344849N	N	3
n407425N	N	3
n437270N	N	3
n192530U	U	5
n283170U	U	5
n398694U	U	5
n400217U	U	5
n79082U	U	5

Table 6: Misclassification Summary for Batch 2

		Predicted Subtype				
		Subtype 1	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
Real Subtype	*Subtype 1*	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

Table 7: Batch 3

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
n29789001	CRF01_AE	2
n29789501	CRF01_AE	2
n31277801	CRF01_AE	2
n51017401	CRF01_AE	2
n51018101	CRF01_AE	2
n149261F1	F1	1
n233666F1	F1	1
n284635F1	F1	1
n312763F1	F1	1

n400218F1	F1	1
n281226G	G	4
n312756G	G	4
n333759G	G	4
n425283G	G	4
n494864G	G	4
n118236N	N	3
n149433N	N	3
n159509N	N	3
n356546N	N	3
n437270N	N	3
n192530U	U	5
n203346U	U	5
n398692U	U	5
n79082U	U	5
n79085U	U	5

Table 8: Misclassification Summary for Batch 3

		Predicted Subtype				
		Subtype 1	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
Real Subtype	*Subtype 1*	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

Table 9 Batch 4

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
n10092302	CRF02_AG	3
n10092802	CRF02_AG	3
n10144602	CRF02_AG	3
n8191802	CRF02_AG	3
n8880302	CRF02_AG	3
n1815204	CRF04_cpx	4

n23145704	CRF04_cpx	4
n23145804	CRF04_cpx	4
n49487804	CRF04_cpx	4
n28317404	CRF04_cpx	4
n11804006	CRF06_cpx	2
n20947906	CRF06_cpx	2
n8880606	CRF06_cpx	2
n9077606	CRF06_cpx	2
n9077706	CRF06_cpx	2
n21666208	CRF08_BC	1
n35360008	CRF08_BC	1
n49557008	CRF08_BC	1
n49560208	CRF08_BC	1
n51175508	CRF08_BC	1
n10091511	CRF11_cpx	5
n22457611	CRF11_cpx	5
n40018711	CRF11_cpx	5
n9077111	CRF11_cpx	5
n9077211	CRF11_cpx	5

Table 10: Misclassification Summary for Batch 4

Predicted Subtype						
Real Subtype		*Subtype 1*	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
	Subtype 1	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

Table 11: Batch 5

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
n118481A	A	3
n118482A	A	3
n118483A	A	3
n157010A	A	3
n157011A	A	3
n159707C	C	4
n177457C	C	4
n38918C	C	4
n52451C	C	4
n89394C	C	4
n100906F2	F2	2
n149262F2	F2	2
n149263F2	F2	2
n203821F2	F2	2
n494848F2	F2	2
n100943G	G	5
n114231G	G	5
n50585G	G	5
n88145G	G	5
n88827G	G	5
n118235N	N	1
n149433N	N	1
n344849N	N	1
n407425N	N	1
n437270N	N	1

Table 12: Misclassification Summary for Batch 5

		Predicted Subtype				
Real Subtype		*Subtype 1*	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
	Subtype 1	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

Table 13: Batch 6

Virus Name	*Los Alamos Subtype*	*Predicted Subtype K-Mean R output*
A1GE1999	A1	4
A1IT2002	A1	4
A1KE2002	A1	4
A1KZ2002	A1	4
A1RU2000	A1	4
BCNRL4	B	2
BFR1983	B	2
BJP2006	B	2
BKR2004	B	2
BUSAC1	B	2
CDK2001	C	1
CGE2003	C	1
CIN2003	C	1
CZA2004	C	1
CZM2002	C	1
DCD1983	D	5
DKR2004	D	5
DTD1999	D	5
DUG1993	D	5
DYE2002	D	5
GBE1996	G	3
GCM2001	G	3
GCU1999	G	3
GES2000	G	3
GPTPT2	G	3

Table 14: Misclassification Summary for Batch 6

		Predicted Subtype				
		Subtype 1	*Subtype 2*	*Subtype 3*	*Subtype 4*	*Subtype 5*
Real Subtype	*Subtype 1*	5				
	Subtype 2		5			
	Subtype 3			5		
	Subtype 4				5	
	Subtype 5					5

Misclassification rate=#of misclassified viruses/total # of viruses=0/25=0

5.2 Classification Results Based on the Individual Gene

See APPENDIX D. The example in Figure D.1 shows nine gene trees for Batch 1. Similarly, the example in Figure D.2 shows nine gene trees for Batch 2 etc. Conservative calculations of the misclassification rate produce the following results. The tables Table 15-26 show classification results and misclassification rate given by the 9 gene trees for each batch.

Table 15: Misclassification Summary for Batch 1

ENV

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					*5*

GAG

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	*5*				
2		*5*			
3			*5*		
4				*5*	
5					*5*

NEF

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

POL

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					*5*

REV

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
1	*5*						
2		*5*					
3			*5*				
4				*3*			*2*
5					*3*	*2*	

TAT

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

VIF

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					*5*

VPR

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*	*6*
1	*5*					
2		*3*				*2*
3			*5*			
4				*5*		
5	*2*		*3*			

VPU

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5	2				3

Table 16: Misclassification rate for Batch 1

Regions	*Misclassification rate*
ENV	0
GAG	0
NEF	0
POL	0
REV	4/25
TAT	0
VIF	0
VPR	7/25
VPU	2/25

Table 17: Misclassification Summary for Batch 2

ENV

	Predicted Subtype						
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5	1				3	1

GAG

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5		2			3

NEF

	Predicted Subtype						
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5		1			3	1

POL

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5		2			3

REV

	Predicted Subtype						
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5	2				2	1

TAT

	Predicted Subtype						
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5					4	1

VIF

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					*5*

VPR

	Predicted Subtype						
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	*5*					
	2		*3*				*2*
	3			*5*			
	4				*5*		
	5	*2*		*3*			

VPU

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5	2				3

Table 18: Misclassification rate for Batch 2

Regions	*Misclassification rate*
ENV	2/25
GAG	2/25
NEF	2/25
POL	2/25
REV	3/25
TAT	1/25
VIF	4/25
VPR	0
VPU	2/25

Table 19: Misclassification Summary for Batch 3

ENV

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5				3	2

GAG

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5		1			4

NEF

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5	1				4

POL

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5		1			4

REV

		Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5	2				2	1

TAT

		Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5					3	2

VIF

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

VPR

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

VPU

		Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*	*6*
	1	5					
	2		5				
	3			5			
	4				5		
	5					3	2

Table 20: Misclassification rate for Batch 3

Regions	*Misclassification rate*
ENV	3/25
GAG	1/25
NEF	1/25
POL	1/25
REV	3/25
TAT	2/25
VIF	0
VPR	0
VPU	2/25

Table 21: Misclassification Summary for Batch 4

ENV

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

GAG

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

NEF

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

POL

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

REV

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

TAT

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

VIF

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					*5*

VPR

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	*5*				
	2		*5*			
	3			*5*		
	4				*5*	
	5					*5*

VPU

	Predicted Subtype					
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

Table 22: Misclassification rate for Batch 4

Regions	*Misclassification rate*
ENV	0
GAG	0
NEF	0
POL	0
REV	0
TAT	0
VIF	0
VPR	0
VPU	0

Table 23: Misclassification Summary for Batch 5

ENV

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

GAG

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

NEF

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

POL

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

REV

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

TAT

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

VIF

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			3		2
	4				5	
	5					5

VPR

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	*5*				
	2		*5*			
	3			*5*		
	4				*5*	
	5					*5*

VPU

		Predicted Subtype				
Real Subtype		*1*	*2*	*3*	*4*	*5*
	1	5				
	2		5			
	3			5		
	4				5	
	5					5

Table 24: Misclassification rate for Batch 5

Regions	*Misclassification rate*
ENV	0
GAG	0
NEF	0
POL	0
REV	0
TAT	0
VIF	2/25
VPR	0
VPU	0

Table 25: Misclassification Summary for Batch 6

ENV

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

GAG

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

NEF

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

POL

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

REV

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

TAT

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4	1			4	
5		1			4

VIF

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*	*6*
1	5					
2		5				
3			5			
4			1	4		
5					4	*1*

VPR

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	*5*				
2		*5*			
3			*5*		
4				*5*	
5					*5*

VPU

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

Table 26: Misclassification rate for Batch 6

Regions	*Misclassification rate*
ENV	0
GAG	0
NEF	0
POL	0
REV	0
TAT	2/25
VIF	2/25
VPR	0
VPU	0

5.3 Classification Results Based on the Concatenation Method

See APPENDIX E. The example in BATCH 1 shows concatenation tree for Batch 1. Similarly, the example in BATCH 2 shows concatenation tree for Batch 2 etc. Conservative calculations of the misclassification rate produce the following results. Table 27 shows the classification given by concatenation method. Table 28 shows misclassification rate for batch 1 through 6.

Table 27: Misclassification Summary for Batches 1-6

BATCH 1

		Predicted Subtype				
		1	*2*	*3*	*4*	*5*
Real Subtype	*1*	5				
	2		5			
	3			5		
	4				5	
	5					5

BATCH 2

		Predicted Subtype				
		1	*2*	*3*	*4*	*5*
Real Subtype	*1*	5				
	2		5			
	3			5		
	4				5	
	5		1			4

BATCH 3

		Predicted Subtype				
		1	*2*	*3*	*4*	*5*
Real Subtype	*1*	5				
	2		5			
	3			5		
	4				5	
	5					5

BATCH 4

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

BATCH 5

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

BATCH 6

Real Subtype \ *Predicted Subtype*	*1*	*2*	*3*	*4*	*5*
1	5				
2		5			
3			5		
4				5	
5					5

Table 28: Misclassification rate for Batch 1-6

Batch	*Misclassification rate*
BATCH 1	0
BATCH 2	1/25
BATCH 3	0
BATCH 4	0
BATCH 5	0
BATCH 6	0

CHAPTER 6

DISCUSSION

Many recent research papers focus on fully automatic Non-Phylogenetic methods to process a large number of sequences. However, it is our position that statistical accuracy should not be sacrificed for the sake of ease or computational speed.

REGA method, for example, applies one single generic model (See Section 2.4) while our method chooses the best fit model for every gene segment for every data sample. As another example of comprehensiveness, our method uses Maximum Likelihood when estimating evolutionary tree. See sections 4.3 and 4.3.1 for explanation of advantages of this approach.

As a tradeoff to speed and wide applicability, automatic methods often produce conflicting results, or are unable to classify new or rare sequences. They sometimes disagree with manually performed phylogenetic analyses. One of the studies that was conducted in 2008 compared 5 popular automatic classification methods. It found that all 5 methods yield different subtypes for new or recombinant forms of HIV.

Table 29: Comparison of Automatic Methods 1

Sample	*Method*				
	STAR	*REGA*	*Geno2pheno*	*jpHMM*	*Virco*
5915–2002b	Not defined	Not defined	CRF10_CD	A1/C/A2	D
5915–2002	Not defined	Not defined	A1	A1/C	U
2566_2005	Not defined	Not defined	A1	A1/C	Not defined
2566_2006	Not defined	Not defined	A1	A1/C	Not defined

(Ntemgwa, 2008)

Another study that was conducted in 2006 in England found that methods agreed poorly, that is agreed in less than 50% for subtypes other than B,C and H. Methods could not classify 5-10% of sequences and returned discordant results in about 12% cases of divergent sequences.

Table 30: Comparison of Automatic Methods 2

STANDORD	*STAR*	*REGA*
High agreement:	B, C , H	
< 50%	Other Subtypes	

(Gifford,2006)

A number of researchers on HIV classification made use of genetic variation of one or two particular genomic region. An example of this approach is in jpHMM algorithm or the classification proposed in (Mezej, 2006).

To evaluate a single gene approach in HIV classification, nine gene trees for each batch were systematically evaluated in chapter 5. We can see different trees support different classifications; hence, the classification is not consistent if researches prioritize one gene over another. By using individual genes, we are limited to partial information yielding contradicting results. As we can see from the result section, the method based on the individual genes gives the highest misclassification rate.

Finally, HIV string is relatively short and it is even shorter if partial information is taken. Therefore, we do not have enough data support to be confident about the classification. We calculated misclassification rate, assuming that the tree is given without taking into the account the gene tree estimation error. To understand the gene tree estimation error we conducted bootstrapping analysis by taking 100 trees and summarizing them using the consensus trees. Bootstrap method gives a measure of a tree robustness and reliability. Any individually produced

tree is a point estimate, while bootstrapping provides a confidence level where bootstrap values tell us how likely a clad will appear with another set of data. Bootstrapping reinforces the weakness of the method because the consensus tree has a low value for some clads meaning the tree produced based on a single gene is not a reliable one; hence, the classification is highly uncertain. For type B HIV sequences, for example, the bootstrapping support is 86% , so the probability is not 100% of them belonging to the same group. A1.KE.2002 has 54% support, so it can be anywhere (See APPENDIX F).

It is clear that phylogenetic analysis based on complete genome is much more reliable than those based on short segments of the HIV-1 genome. Moreover, in some cases classification could not be resolved if only part of the sequence region is utilized in classification.

However, further investigations also revealed errors in classification based on the whole genome, the concatenation method. When one single model is chosen for modeling a phylogenetic tree, the assumption is made that each genome region undergoes the same evolutionary change. However, as mentioned in Introduction, this assumption is not accurate for HIV viruses. When we build one single tree based on the entire super gene, we assume that all nine trees are the same. However, if trees for the individual gene are not the same, our assumption stretches too much, leading to inaccurate classification. The Batch 2 had demonstrated the above discussion by misclassifying one of HIV strings.

We have observed the significant improvement in HIV string classification by utilizing information provided by the entire genome. Therefore, in our method we take advantage of the whole genome, and at the same time recognizing the uniqueness and contribution of every genome region. With our method, we achieved 100% subtyping accuracy.

REFERENCES

[1] Abecasis, A., Vandamme, A.M., Lemey, P. 2006 Sequence alignment in HIV computational analysis, *Los Alamos National Laboratory, Los Alamos, New Mexico*, 87545

[2] Bos, D.H., Posada, D. 2005 Using models of nucleotide evolution to build phylogenetic trees *Developmental and Comparative Immunology* **29** 211–227

[3] Bushman, F., Nabel, G., Swanstrom, R., 2012 HIV : from biology to prevention and treatment. *Cold Spring Harbor perspectives in medicine*; **1-90** 184-190

[4] Castro-Nallar E. *et al*. 2012 The evolution of HIV: inferences using phylogenetics. *Mol Phylogenet Evol.,* **62(2)**, 777-792.

[5] Darriba, D., Taboada, G.L., Doallo, R., Posada, D. 2012. jModelTest 2: more models, new heuristics and parallel computing. *Nature Methods* **9**, 772

[6] d'Ettorre, G. *et al*. 2010 The role of HIV-DNA testing in clinical practice. New Microbiol., **33**, 1-11

[7] Dwivedi, S.K. *et al*. 2012 Classification of HIV-1 sequences psing profile hidden Markov models. *PLoS ONE,* **7(5)**, e36566.

[8] Feller, W. 1968 An introduction to probability theory and its applications. *New York: Wiley*

[9] Felsenstein, J. 1973 Maximum-likelihood estimation of evolutionary trees from continuous characters. *Am. J. Hum. Genet.* **25**:471-492

[11] Felsenstein, J. 1978 Cases in which Parsimony or Compatibility Methods will be Positively Misleading *Syst Zool* **27:**401-410

[12] Felsenstein, J. 1981 Evolutionary trees from DNA sequences: a maximum likelihood approach *J. Mol. Evol.* **17**:368-376

[13] Gifford, R., de Oliveira, T., Rambaut, A., Myers, R.E., Gale, C.V., *et al.* 2006 Assessment of automated genotyping protocols as tools for surveillance of HIV-1 genetic diversity. *AIDS* **20**: 1521–1529.

[14] Gomez-Carrillo, M. *et al.* 2004 Drug resistance testing provides evidence of the globalization of HIV type 1: a new circulating recombinant form. *AIDS Res. Hum. Retrovir. PubMed.,* **20,** 885-888.

[15] Guindon, S., Gascuel, O. 2003 A simple, fast and accurate method to estimate large phylogenies by maximum-likelihood". *Systematic Biology* **52**: 696-704.

[16] Hartl, D., Jones, E., 2001 Genetics : analysis of genes and genomes. *Jones and Bartlett*

[17] Hasegawa, M., Kishino, H., Saitou, N. 1991 On the maximum likelihood method in molecular phylogenetics. *J.Mol. Evol.,* **32:**443-445

[18] Holquin, A., Lopez, M. , Soriano, A. 2008, Reliability of rapid subtyping tools compared to that of phylogenetic analysis for characterization of human immunodeficiency virus type 1 non-B subtypes and recombinant forms. *J. Clin. Microbiol* **46**(12):3896-3899.

[19] Huet T, Cheynier R, Meyerhans A, *et al.* 1990 Genetic organization of a chimpanzee lentivirus related to HIV-1**.** *Nature* **345:**356-9

[20] Kim, J**.** *et al.* 2010 A classification approach for genotyping viral sequences based on multidimensional scaling and linear discriminant analysis. *Bioinformatics,* **11**:434.

[21] Korber, B., Gaschen, B. *et al.* 2001 Evolutionary and immunological implications of contemporary HIV-1 variation *Br. Med. Bull* **58**:19-42.

[22] Kuiken, C. *et al*. 2003 HIV sequence database. PubMed., **5(1)**, 52-61.

[23] Kuiken, C., Foley, B. *et al.* 2011 HIV sequence compendium 2011, *Los Alamos National Laboratory, Los Alamos, New Mexico* 87545

[24] Leitner, T. *et al.* 1996 Accurate reconstruction of a known HIV-1 transmission history by phylogenetic tree analysis. *PNAS.*, **93 (20)**, 10864-10869.

[25] Leitner T. *et al.* 2005 HIV-1 subtype and circulating recombinant form (CRF) reference sequences *Los Alamos National Laboratory, Los Alamos, NM* 87545

[26] Li, G. *et al.* 2009 A comparative analysis of biclustering algorithms for gene expression data *Nucl. Acids Res.*, **37(15)**, e101.

[27] Li, M. *et al.* 2011 DNA requirements for assembly and stability of HIV-1 intasomes. Protein Sci., **21(2)**, 249–257.

[28] Liu,L., Yu,L., Kubatko, L., Pearl, D.K., Edwards, S. V., 2009 Coalescent methods for estimating phylogenetic trees. *Molecular Phylogenetics and Evolution* **53:** 320–328

[29] Liu, L., Yu, L. 2011 Estimating Species Trees from Unrooted Gene Trees *Syst. Biol.* **60**: 661-667.

[30] Lole, K.C., Bollinger, R. C., 1999 Full-length human immunodeficiency virus type 1 genomes from subtype C-Infected seroconverters in India, with evidence of intersubtype recombination, *J. Virol.* vol. **73,** no. 1, 152-160

[31] Mansky, L. M., Temin, H.M. 1995 Lower in vivo mutation rate of human immunodeficiency virus type 1 than that predicted from the fidelity of purified reverse transcriptase *J Virol.* **69(8)**: 5087–5094.

[32] Mays, L.L., 1981 Genetics : a molecular approach. *Macmillan*

[33] Merrell, D. 1975 Introduction to genetics. *Norton*

[34] Mezei, M., Balog, K. *et al*. 2006 Genetic variability of gag and env regions of HIV type 1 strains circulating in Slovenia *AIDS Res Hum Retroviruses.* **22(1):**109-113.

[35] Moret, B.M.E. *et al.* 2004 Phylogenetic Networks: Modeling, Reconstructability, and Accuracy *IEEE/ACM Trans. Comput. Biology Bioinform.* **1(1):** 13-23

[36] Myers,R.E. *et al.* 2005 A statistical model for HIV-1 sequence classification using the subtype analyser (STAR) *Bioinformatics,* **21**,3535–3540.

[37] Neill, D.B., 2011 Fast Bayesian scan statistics for multivariate event detection and visualization. *Statistics in medicine,* Vol. **35,** issue 5, 455-469

[38] Ntemgwa, M., Gill, M.J., Brenner, B.G., Moisi, D., Wainberg, M.A. 2008 Discrepancies in assignment of subtype/recombinant forms by genotyping programs for HIV type 1 drug resistance testing may falsely predict superinfection. *AIDS Res Hum Retroviruses* **24**: 995–1002

[39] Paraskevis, D., Deforche, K., Lemey, P. *et al.*, 2005 SlidingBayes: exploring recombination using a sliding window approach based on Bayesian phylogenetic inference. *Bioinformatics.* **21**(7):1274-1275

[40] Pond, S.L.K., Posada, D., Stawiski, E. 2009 An evolutionary model-based algorithm for accurate phylogenetic breakpoint mapping and subtype prediction in HIV-1.*PLoS Comput Biol.;* ***5***(11)

[41] Posada, D., Crandall, K.A., 2001 Evaluation of methods for detecting recombination from DNA sequences: Computer simulations. *Proc Natl Acad Sci* **98**(24): 13757–13762

[42] Posada, D., Buckley, T.D. 2004 Model selection and model averaging in phylogenetics: advantages of akaike information criterion and bayesian approaches over likelihood ratio tests *Syst. Biol.* **53(5)**:793-808

[43] Rambaut,A. *et al.* 2004 The causes and consequences of HIV evolution. *Nat. Rev. Gene.,* **5**, 52-61

[44] Rogers, J. S. 1997 On the consistency of maximum likelihood estimation of phylogenetic trees from nucleotide sequences. *Syst. Biol*. **46:**354–357

[45] Rozanov, M. *et al.* 2004 A web-based genotyping resource for viral sequences. *Nucleic Acids Res.*, **32**, W654–9.

[46] Salminen, M., Carr, J.K., Burke, D.S., and McCutchan, F.E. 1995 Identification of recombination breakpoints in HIV-1 by bootscanning. *Laboratory of Tumor Cell Biology National Cancer Institute Meeting, Bethesda, MD*

[47] Salminen, M.O., Carr, J.K., Burke,D.S., McCutchan, F.E., 1995 Genotyping of HIV-1, *Los Alamos National Laboratory, Los Alamos, NM,* 87545

[48] Schochetman, G., George R., 1992 AIDS testing : methodology and management issues *Springer-Verlag*

[49] Schochetman, G., George R., 1994 AIDS testing :a comprehensive guide to technical, medical, social, legal, and management issues *Springer-Verlag*

[50] Schultz, B. *et al.* 2010 HIV classification using the coalescent theory. *Bioinformatics*, **26**, 1409.

[51] Sul, S.J., Williams,T.L. 2008 An experimental analysis of Robinson-Foulds distance matrix algorithms *Proc. of the 16th annual European symposium on Algorithms* pp. 793-804

[52] Sullivan, J., Joyce, P. 2005 Model selection in phylogenetics. *Annu. Rev. Ecol. Evol. Syst.* **36**:445–66

[53] Svarovskaia, E.S., Zhang, X. *et al.* 2003 Azido-containing aryl beta-diketo acid HIV-1 integrase inhibitors *Bioorg Med Chem Lett* **13(6):**1215-1219.

[54] Triques, K., Bourgeois, A., Vidal, N. 2000, Near-full-length genome sequencing of divergent african HIV type 1 subtype F viruses leads to the identification of a new HIV type 1 subtype designated K, *Aids research and human retroviruses* Volume **16,** Number 2**,** 139-151

[55] Wu, X. *et al.* 2004 Whole Genome Phylogeny via Complete Composition Vectors

[56] Wu, X. *et al.* 2007 Nucleotide composition string selection in HIV-1 subtyping using whole genomes. *Bioinformatics*, **23**, 1744–1752

[57] Yang, Z., 1994 Maximum-likelihood estimation of phylogeny from DNAsequences when substitution rates differ over sites. *Mol. Biol. Evol.* **10(6):** 1396- 1401

[58] A Tutorial on Clustering Algorithms. Clustering: An Introduction http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/

[59] A Tutorial on Clustering Algorithms. K-Means Clusteringhttp://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html

[60] Andrew Moore. K-means and Hierarchical Clustering - Tutorial Slides. http://www-2.cs.cmu.edu/~awm/tutorials/kmeans.html

[61] phybase an R package for phylogenetic analysis retrieved May 25, 2013 from code.google.com/p/phybase

[62] Centers for Disease Control and Prevention. Monitoring selected national HIV prevention and care objectives by using HIV surveillance data—United States and 6 U.S. dependent areas—2010. *HIV Surveillance Supplemental Report* 2012;17(No. 3, part A). http://www.cdc.gov/hiv/topics/ surveillance/resources/reports/. Published June 2012. Accessed 2013

[62] UNAIDS. UNAIDS World AIDS Day Report 2012. http://www.unaids.org/en/media/unaids/contentassets/documents/epidemiology/2012/gr2012/JC2434_WorldAIDSday_results_en.pdf. Published 2013. Accessed 2013

APPENDIX A: DATA

Table 31: Downloaded Sequence Data for Batch 1

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
DQ366662	05MYKL045_1	33_01B	Malaysia	2005	9153	n9014933
DQ366660	05MYKL015_2	33_01B	Malaysia	2005	9062	n9015133
DQ366659	05MYKL007_1	33_01B	Malaysia	2005	9081	n9015233
AB547464	JKT194-C	33_01B	Indonesia	2007	8857	n34227533
AB547463	JKT189-C	33_01B	Indonesia	2007	8936	n34227633
EF158041	05AF095	35_AD	Afghanistan	2005	8703	n6609735
GQ477445	073H	35_AD	Afghanistan	2006	8694	n35058735
GQ477442	077H	35_AD	Afghanistan	2006	8697	n35059035
AB703615	10IR.THR41F	35_AD	Iran	2010	8665	n49675335
AB703612	11IR.KSH31F	35_AD	Iran	2011	8671	n49676035
AB287379	93RW037A	A1	Rwanda	1993	9073	n118481A
JX236678	pR880F	A1	Rwanda	2007	9740	n118482A
AM000055	02CD_KTB035	A1	Congo	2002	9740	n118483A
AY521631	DDJ369	A1	Senegal	2001	8861	n157009A
AY521630	DDJ360	A1	Senegal	1996	8777	n157010A
AY371158	02CM_0016BBY	F2	Cameroon	2002	8349	n100906F2
AJ249236	95CM-MP255	F2	Cameroon	1995	8555	n149263F2
AF377956	CM53657	F2	Cameroon	1997	8782	n203821F2
AF247520	CA16	F2	Cameroon	1993	8936	n494848F2
JX140672	DEMF210CM001	F2	Cameroon	2010	8907	n494866F2
AY169816	98CMA105	O	Cameroon	1998	9195	n89413O
AY169815	99CMU4122	O	Cameroon	1999	9186	n89414O
AY169811	98CMABB197	O	Cameroon	1998	9072	n89418O
AY169805	97US08692A	O	United States	1997	9147	n89424O
AY623602	pCMO2_5	O	Cameroon	2004	9860	n159169O

Table 32: Downloaded Sequence Data for Batch2

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
EF029069	U_NL_01_H10986_C11	U	Netherlands	2001	8670	n79082U
AY046058	99GR303	U	Congo	1999	8909	n192530U
FJ388921	CY090	U	Cyprus	2005	8149	n283170U
HM215249	TV721	U	Canada	1999	9788	n398694U
JF683772	CY223	U	Cyprus	2008	8227	n400217U
AJ271370	YBF106	N	Cameroon	1997	9045	n148891N
AY532635	DJO0131	N	Cameroon	2002	8938	n159509N
GQ324959	SJGddd	N	Cameroon	2002	8615	n344849N
GQ324961	02CM_TIM0217	N	Cameroon	2002	9183	n407425N
FB675267	YBF30_patent	N	Cameroon	1995	8892	n437270N
AY586549	Cu87	G	Cuba	1999	9185	n50585G
U88826	92NG083_JV10832	G	Nigeria	1992	9655	n88145G
AB287004	03GH175G	G	Ghana	2003	9725	n88826G
AF450098	X138	G	Spain	2000	8739	n89440G
AB231893	GHNJ175	G	Ghana	2003	9724	n101448G
DQ979024	X1670	F1	Spain	2007	8951	n62007F1
AY173958	BZ163	F1	Brazil	1989	8991	n181594F1
FJ900268	AO_06_ANG58	F1	Angola	2006	8240	n284633F1
FJ771010	07BR844	F1	Brazil	2007	9356	n288671F1
AB485658	BCI_R07	F1	Romania	1996	9747	n312761F1
EF036533	Fj064	01_AE	China	2006	9732	n7766301
AY125894	97TH6_107	01_AE	Thailand	1997	8832	n8943801
AB253680	DR3730	01_AE	Japan	2006	8654	n10258001
AY945716	00TH_C2101	01_AE	Thailand	2000	8371	n10622801
AY713425	90TH_CM244	01_AE	Thailand	1990	8796	n14200001

Table 33: Downloaded Sequence Data for Batch 3

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
EF029068	U_NL_01_H10986_D1	U	Netherlands	2001	9670	n79082U
EF029066	U_NL_95_H10986_D1	U	Netherlands	1995	9670	n79085U
AF457101	90CD121E12	U	Congo	1990	8784	n192530U
AF286236	83CD003_Z3	U	Congo	1983	9060	n203346U
HM215251	TV749	U	Canada	2001	9788	n398692U
DQ017382	04CM_1015_04	N	Cameroon	2004	8926	n118236N
DQ017382	04CM_1015_04	N	Cameroon	2004	9182	n149433N
GQ324958	U14842	N	Cameroon	2006	8938	n159509N
FB675251	YBF30_patent	N	Cameroon	1995	9183	n356546N
JN572926	N1_FR_2011	N	France	2011	8892	n437270N
FJ389363	178_15	G	Cameroon	2004	9096	n281226G
AB485663	HH8793	G	Finland	1993	9707	n312756G
FJ670520	X1628-2	G	Spain	2005	9018	n333759G
JN248591	09NG_SC31	G	Nigeria	2009	8837	n425283G
JX140676	DEMG10CM008	G	Cameroon	2010	8979	n494864G
AJ249238	96FR-MP411	F1	France	1996	8614	n149261F1
AF005494	93BR020_1	F1	Brazil	1993	8968	n233666F1
FJ900266	AO_06_ANG32	F1	Angola	2006	8257	n284635F1
AB485656	BZ163	F1	Brazil	1990	9602	n312763F1
JF683771	CY222	F1	Cyprus	2008	8096	n400218F1
FJ185237	97VNHCM301	01_AE	Viet nam	1997	8842	n29789001
FJ185232	98VNHD9	01_AE	Viet nam	1998	8854	n29789501
AB485652	ID17	01_AE	Indonesia	1993	9648	n31277801
JX960618	09LNA041	01_AE	China	2009	8404	n51017401
JX960604	09LNA008	01_AE	China	2009	8359	n51018101

Table 34: Downloaded Sequence Data for Batch 4

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
AB231895	GHNJ185	02_AG	Ghana	2003	9715	n8191802
AB286852	03GH173_06	06_cpx	Ghana	2003	9793	n8880606
AB286855	03GH181AG	02_AG	Ghana	2003	9724	n8880302
AB746344	p00CH-WS035_08_BC51	08_BC	China	2000	9646	n21666208
AB746345	p00CH-WS035_08_BC52	08_BC	China	2000	9634	n35360008
AB773885	p00CH-HH090_08_BC30	08_BC	China	2000	9598	n49557008
AF049337	94CY032_3	04_cpx	Cyprus	1994	9050	n1815204
AF064699	BFP90	06_cpx	Australia	1996	9775	n9077606
AF119819	GR84_97PVMY	04_cpx	Greece	1997	9699	n23145704
AF119820	GR11_97PVCH	04_cpx	Greece	1991	9548	n23145804
AF179368	GR17	11_cpx	Greece	2000	8935	n9077111
AJ288981	97SE1078	06_cpx	Senegal	1997	9808	n9077706
AJ288982	95ML127	06_cpx	Mali	1995	9719	n11804006
AJ291718	MP818	11_cpx	Cameroon	1997	9711	n9077211
AJ291720	MP1307	11_cpx	France	1999	9769	n10091511
AY008715	97CNGX_6F	08_BC	China	1997	8802	n49560208
AY371136	01CM_1237NG	02_AG	Cameroon	2001	8377	n10092302
AY371141	02CM_4082STN	02_AG	Cameroon	2002	8413	n10092802
AY371149	01CM_0186ND	11_cpx	Cameroon	2001	8409	n22457611
DD409979	IbNG-patent	02_AG	Nigeria	2007	9201	n10144602
DQ400856	04RU001	06_cpx	Russia	2005	9224	n20947906
FJ388917	CY081	04_cpx	Cyprus	2005	8214	n28317404
HM067748	nx2	08_BC	China	2006	9680	n51175508
JF683802	CY259	11_cpx	Cyprus	2009	8176	n40018711
JX140648	DE00400GR002	04_cpx	Greece	2000	9007	n49487804

Table 35: Downloaded Sequence Data for Batch 5

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
AB287003	03GH175G	G	Ghana	2003	9707	n50585G
GQ344966	06CM06BDHS024	F2	Cameroon	2006	8782	n100906F2
AJ006022	YBF30	N	Cameroon	1995	9182	n118235N
GQ432919	CMNYU124_0_1	F2	Cameroon	2001	8555	n149262F2
AJ249237	95CM-MP257	F2	Cameroon	1995	8589	n149263F2
AM000053	97CD_KCC2	A1	Congo	1997	9740	n118481A
AM000054	97CD_KTB13	A1	Congo	1997	9073	n118482A
AY322184	ML013_10	A1	Kenya	1986	9740	n118483A
AY253322	BD9_11	C	Tanzania	2001	8819	n38918C
AB485662	HH8793	G	Kenya	1993	8367	n88145G
GU332509	P2059b12	F2	Spain	2008	8349	n203821F2
AY521629	DDI579	A1	Senegal	2001	8801	n157010A
AB253422	92RW008	A1	Rwanda	1992	8777	n157011A
AY586547	Cu74	G	Cuba	1999	9185	n88827G
EU786670	P962	G	Spain	2005	9655	n100943G
AB254148	02ZM115	C	Zambia	2002	9076	n52451C
DQ017383	04CM_1131_03	N	Cameroon	2004	8975	n149433N
DQ168576	01NGPL0669	G	Nigeria	2001	8837	n114231G
AF110969	96BW1104	C	Bostwana	1996	8897	n89394C
AF110971	96BW11B01	C	Bostwana	1996	8908	n159707C
GQ324962	U14296	N	Cameroon	2006	8545	n344849N
HV199807	YBF30_patent	N	Cameroon	1995	9183	n407425N
JN572926	N1_FR_2011	N	France	2011	8892	n437270N
JX140673	DEMF210CM007	F2	Cameroon	2010	8936	n494848F2
AB097871	mIDU101_3	C	Myanmar	1999	9073	n177457C

Table 36: Downloaded Sequence Data for Batch 6

Accession	*Los Alamos Name*	*Subtype*	*Country*	*Year*	*Sequence Length*	*Name*
EU861977	60000	A1	Italy	2002	9781	A1.IT.2002.
DQ207944	99GEMZ011	A1	Georgia	1999	8792	A1.GE.1999.
EU110097	ML1990PCR	A1	Kenya	2002	8942	A1.KE.2002.
EF545108	RU00051	A1	Russia	2000	8806	A1.RU.2000.
EF589044	02KZPAV300502	A1	Kazakstan	2002	8810	A1.KZ.2002.
X01762	REHTLV3_LAI_IIIB	B	France	1983	9748	B.FR.1983.
U71182	RL42	B	China	2008	8985	B.CH.-.RL4.
AB428562	574	B	Japan	2006	8698	B.JP.2006.
DQ295195	04KJin8_1955	B	South Korea	2004	9402	B.KR.2004.
EU616645	AC160_T7_Day_762_Dom	B	United States	2008	8606	B.US.-.AC1.
DQ207941	03GEMZ033	C	Georgia	2003	8835	C.GE.2003.
EF469243	D24	C	India	2003	9830	C.IN.2003.
EF514713	CTL_015	C	Denmark	2001	8857	C.DK.2001.
DQ011180	04ZASK202B1	C	South Africa	2004	9076	C.ZA.2004.
AB254153	02ZMDB	C	Zambia	2002	9723	C.ZM.2002.
AY795907	02YE516	D	Yemen	2002	8797	D.YE.2002 .
A07108	ELI_patent	D	Congo	1983	9176	D.CD.1983.
AF084936	DRCBL	G	Belgium	1996	9707	G.BE.1996.
AF423760	X558	G	Spain	2000	8950	G.ES.2000.
AJ519489	MN012	D	Chad	1999	8850	D.TD.1999.
AY371121	01CM_4049HAN	G	Cameroon	2001	8367	G.CM.2001.
AY586549	Cu87	G	Cuba	1999	9185	G.CU.1999
AY612637	PT2695	G	Portugal	2007	9655	G.PT.-.PT2.
AY713418	93UG_065	D	Uganda	1993	8804	D.UG.1993.
DQ054367	04KBH8	D	South Korea	2004	9490	D.KR.2004.

Table 37: Nucleotide summary statistics for Batches 1-6

	GAG	*POL*	*ENV*	*TAT*	*REV*	*VIF*	*VPR*	*VPU*	*NEF*
Batch 1									
Nucleotide	*Counts*								
A	13681	29042	22574	2424	2686	5282	2519	2369	5008
T	7160	16355	15548	1430	1712	3188	1641	1597	3206
C	7270	12464	11416	1833	2088	2498	1326	600	3090
G	9236	17320	15056	1933	2615	3493	1841	1599	4090
	Gene Length								
	1593	3048	2934	312	432	582	315	294	681
Batch 2									
Nucleotide	*Counts*								
A	13601	29354	22512	2531	2742	5288	2488	2305	4537
T	7229	16346	15622	1417	1680	3182	1614	1495	2778
C	7322	12513	11108	1789	2052	2512	1331	600	2829
G	9398	17157	14984	1923	2622	3486	1827	1623	3757
	Gene Length								
	1725	3132	2889	315	381	582	291	315	717
Batch 3									
Nucleotide	*Counts*								
A	13631	29367	22302	2540	2748	5319	2484	2310	4748
T	7214	16321	15557	1426	1655	3173	1616	1533	2910
C	7366	12504	10999	1783	2062	2518	1339	576	2936
G	9370	17202	14957	1904	2593	3458	1811	1637	3884
	Gene Length								
	1704	3123	2766	312	378	582	291	306	699
Batch 4									
Nucleotide	*Counts*								
A	13753	29426	22631	2456	2655	5260	2479	2331	4643
T	7121	16175	15768	1420	1635	3224	1625	1559	2814
C	7246	12578	11082	1853	2079	2482	1332	602	2851
G	9301	17153	14975	1941	2607	3517	1870	1655	3854
	Gene Length								
	1641	3138	2772	321	378	585	315	264	690
Batch 5									
Nucleotide	*Counts*								
A	13625	29276	22438	2526	2740	5243	2539	2356	4809
T	7179	16343	15659	1449	1668	3238	1578	1571	3004
C	7399	12507	11167	1757	2060	2499	1347	581	3011
G	9348	17261	15098	1914	2648	3511	1802	1636	3955
	Gene Length								
	1695	3117	2895	306	372	585	294	333	675
Batch 6									
Nucleotide	*Counts*								
A	13824	29346	22645	2449	2646	5256	2471	2404	4892
T	7131	16322	15627	1443	1681	3227	1610	1538	3089
C	7342	12445	11118	1844	2102	2501	1368	633	3144
G	9137	17157	15068	1908	2656	3489	1824	1643	4187
	Gene Length								
	1608	3075	2805	311	375	582	297	289	676

APPENDIX B: PROCESS FLOW

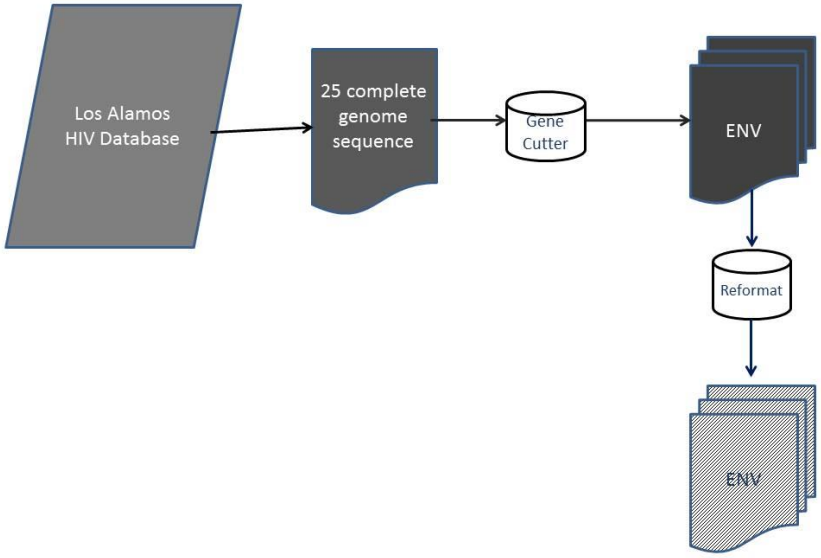


Figure B.1: Data Preprocessing

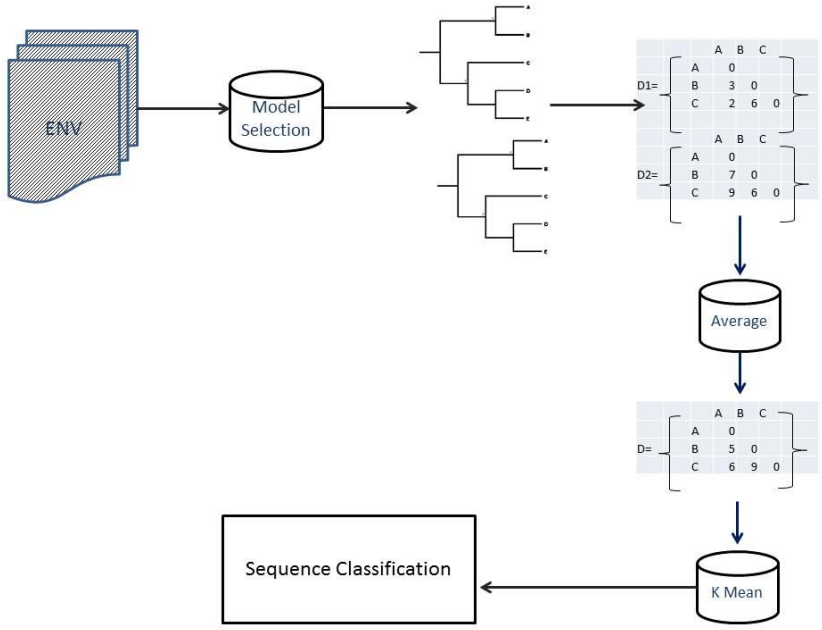
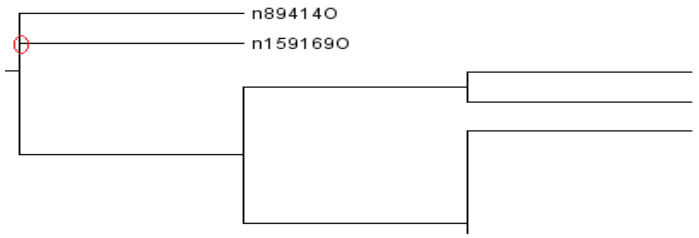


Figure B.2: Data Analysis

APPENDIX C: DISTANCE MATRIX

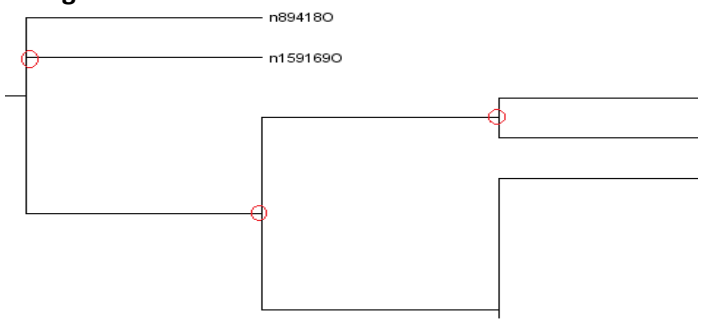
ENV Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0	1	3	3	3
n1591690	1	0	3	3	3
n89420	3	3	0	3	3
n894130	3	3	3	0	1
n894180	3	3	3	1	0



POL Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	3	0	Symmetric		
n89420	1	3	0		
n894130	3	3	3	0	
n894180	3	1	3	3	0



VIF Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	2	0	Symmetric		
n89420	1	2	0		
n894130	3	2	3	0	
n894180	4	3	4	2	0

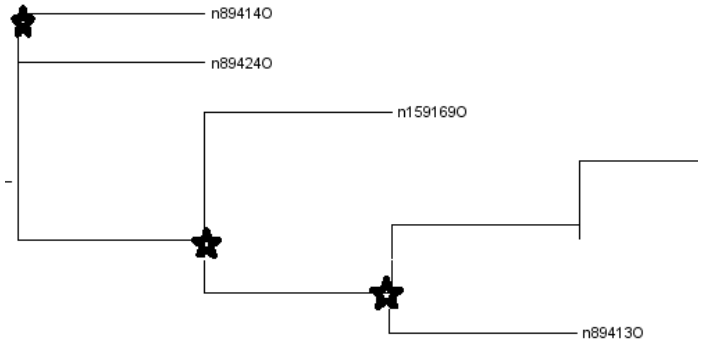
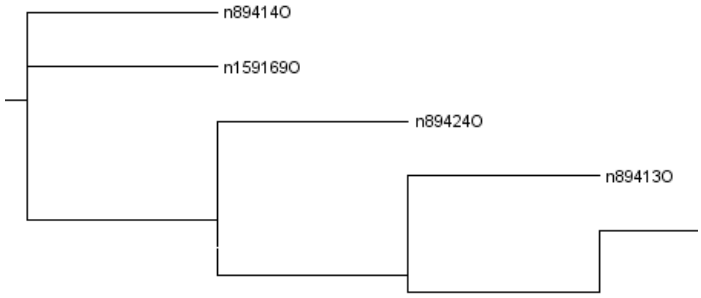


Figure C.1: Distance Matrix Calculation for 9 HIV Genome Regions (continue)

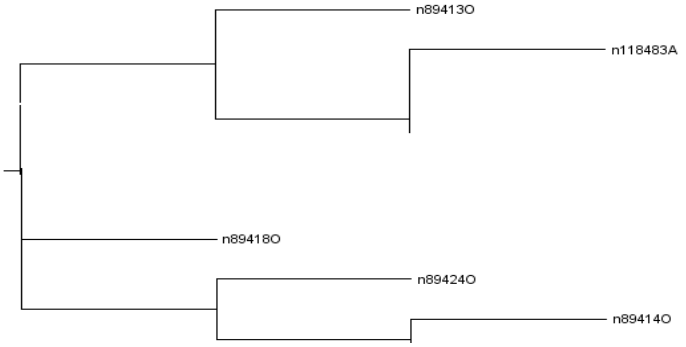
NEF Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0	1	2	3	4
n1591690	1	0	2	3	4
n89420	2	2	0	2	3
n894130	3	3	2	0	2
n894180	4	4	3	2	0



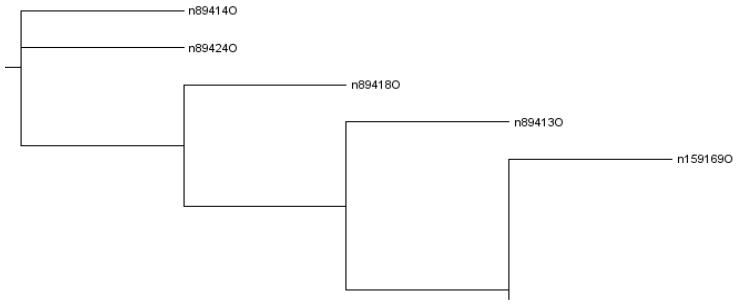
VPU region

	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	1	0	Symmetric		
n89420	2	2	0		
n894130	4	4	3	0	
n894180	3	3	2	2	0



GAG Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0	4	1	3	2
n1591690	4	0	4	2	3
n89420	1	4	0	3	2
n894130	3	2	3	0	2
n894180	2	3	2	2	0



REV Region

	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	2	0	Symmetric		
n89420	3	2	0		
n894130	4	3	2	0	
n894180	1	2	3	4	0

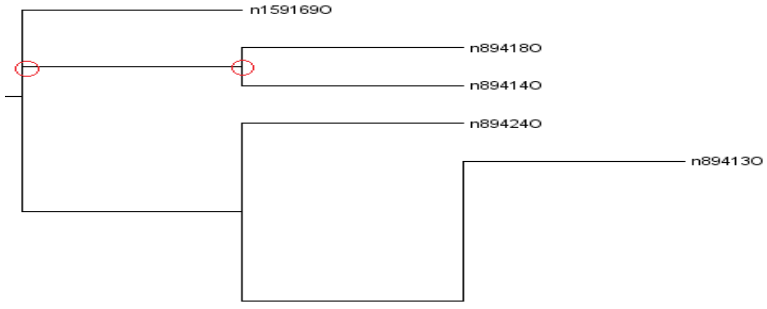
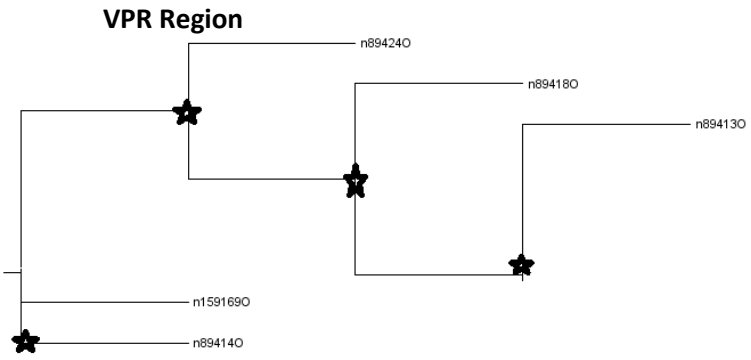


Figure C.1: Distance Matrix Calculation for 9 HIV Genome Regions (continue)

	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	1	0	Symmetric		
n89420	2	2	0		
n894130	4	4	3	0	
n894180	3	3	2	2	0



	n894140	n1591690	n89420	n894130	n894180
n894140	0				
n1591690	2	0	Symmetric		
n89420	1	2	0		
n894130	4	3	4	0	
n894180	3	2	3	2	0

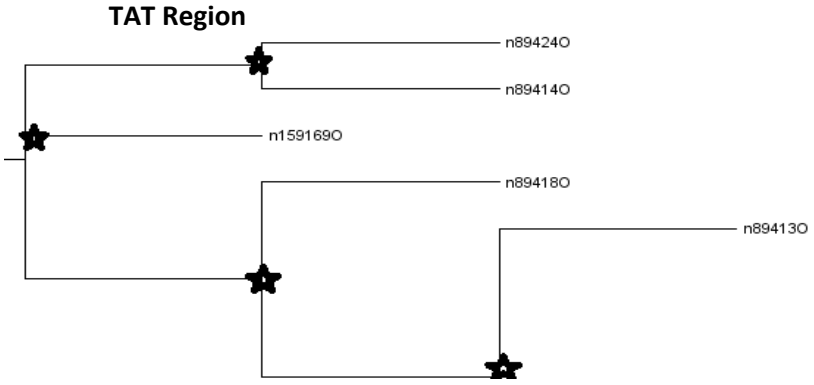


Figure C.1: Distance Matrix Calculation for 9 HIV Genome Regions

The distance between two leaves in a tree is equal to the number of internodes between two leaves in the tree. The internodes between n894140 and n1591690 in the gene tree for *pol* region are highlighted in red. The distance between n894140 and n1591690 is 3 for *pol* Tree and 2 for *rev* Tree.

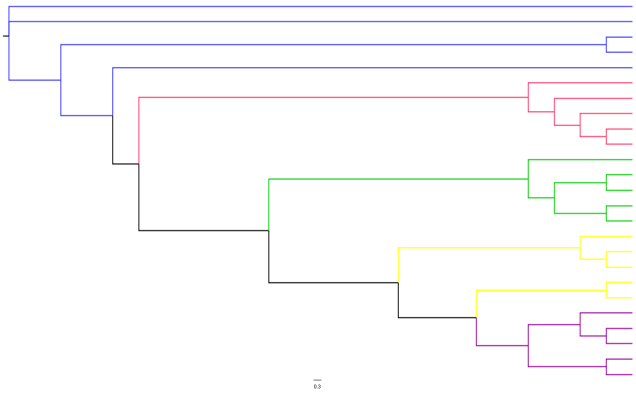
Full Genome

	n894140	n159190	n89420	n894130	n894180
n894140	0				
n1591690	(1+4+1+3+2+2+2+1+1)/9	0	Symmetric		
n89420	(3+1+2+1+3+1+1+2+2)/9	(3+4+2+3+2+2+2+2+2)/9	0		
n894130	(3+3+3+3+4+4+3+4+4)/9	(3+2+3+3+3+3+2+4+4)/9	(3+3+2+3+2+4+3+3+3)/9	0	
n894180	(3+2+4+3+1+3+4+3+3)/9	(3+3+4+1+2+2+3+3+3)/9	(3+2+3+3+3+3+4+2+2)/9	(1+2+2+3+4+2+2+2+2)/9	0

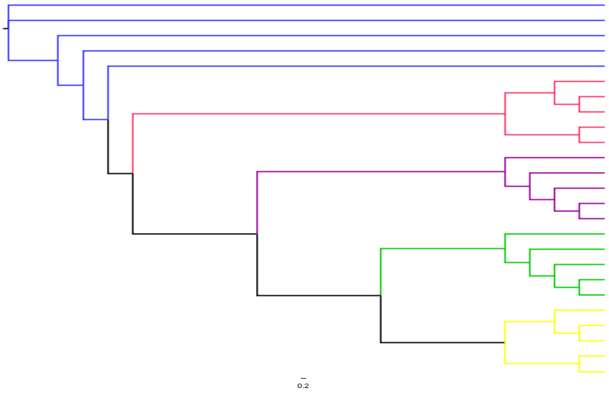
	n894140	n159190	n89420	n894130	n894180
n894140	0				
n1591690	1.889	0	Symmetric		
n89420	1.778	2.444	0		
n894130	3.444	3.000	2.889	0	
n894180	2.889	2.667	2.778	2.222	0

Figure C.2: Distance Matrix of the Average Distances

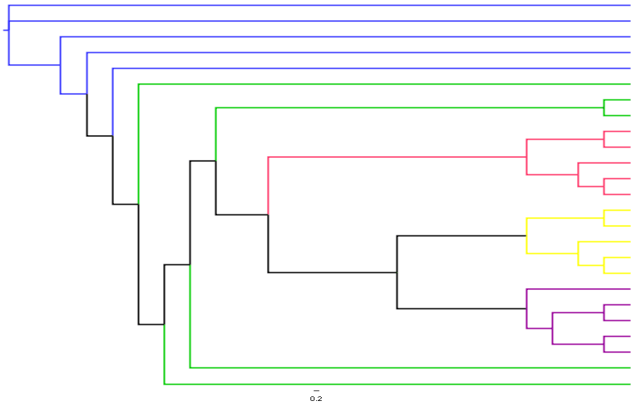
APPENDIX D: GENE TREES



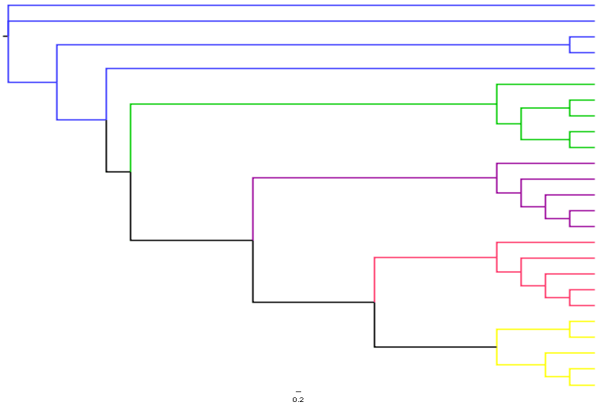
ENV



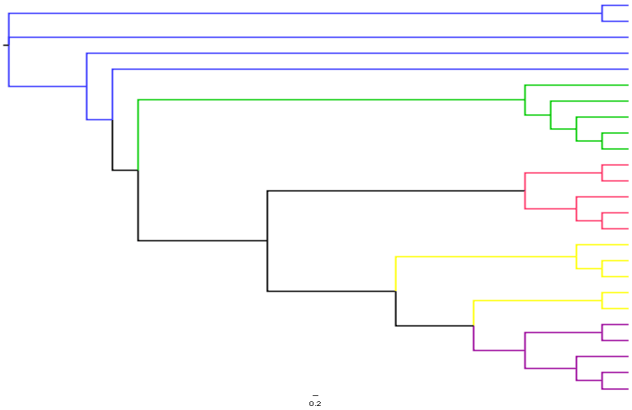
GAG



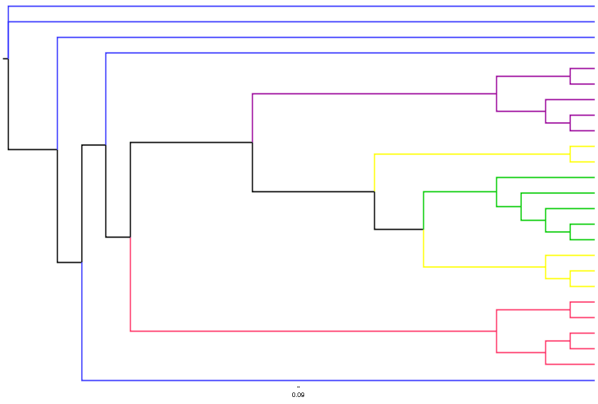
NEF



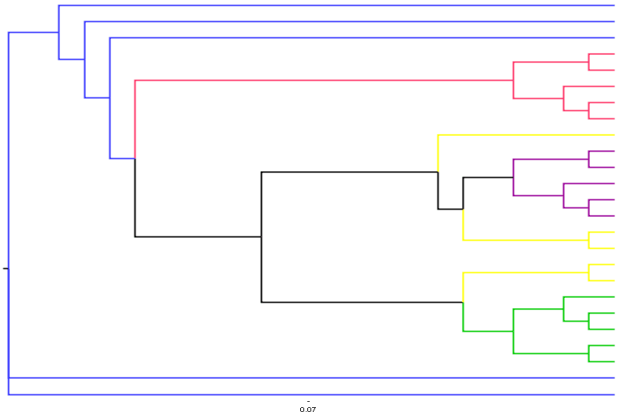
POL



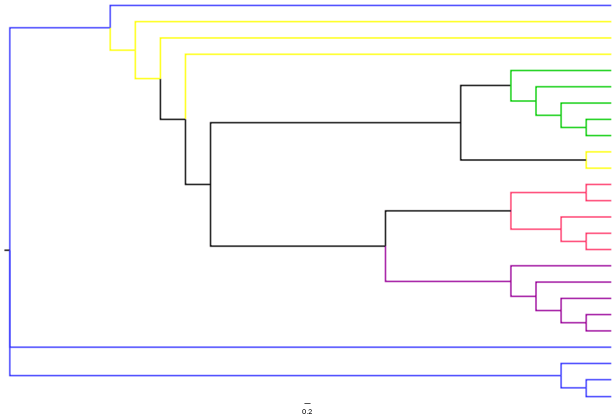
TAT



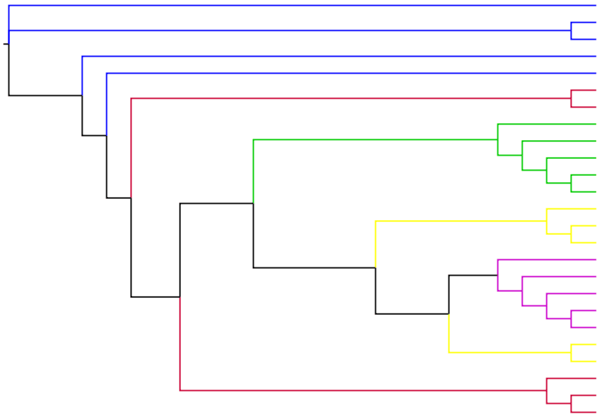
VIF



VPR



VPU



REV

Figure D.1: Gene Trees for Batch 1. Individual Gene Classification Method

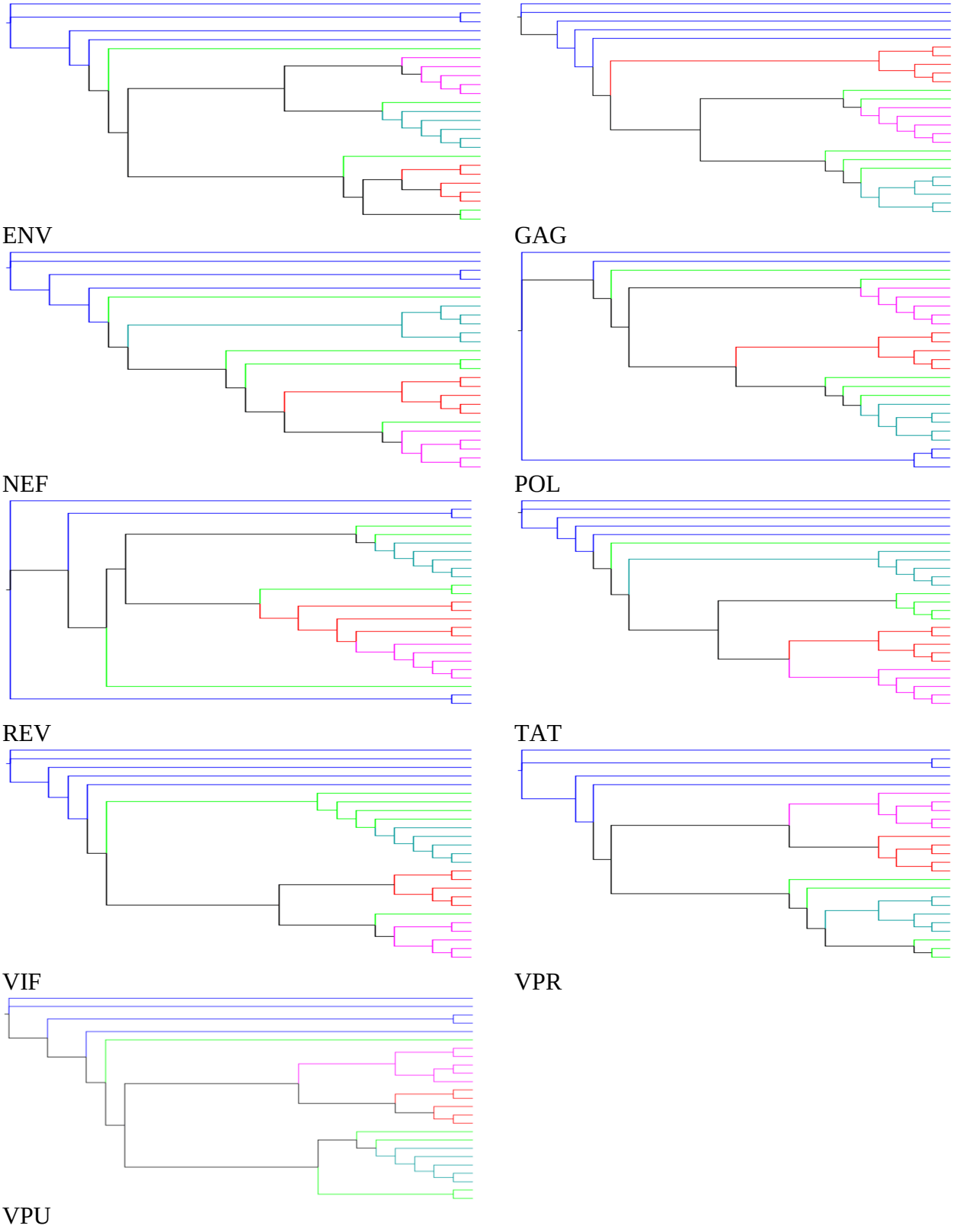


Figure D.2: Gene Trees for Batch 2. Individual Gene Classification Method

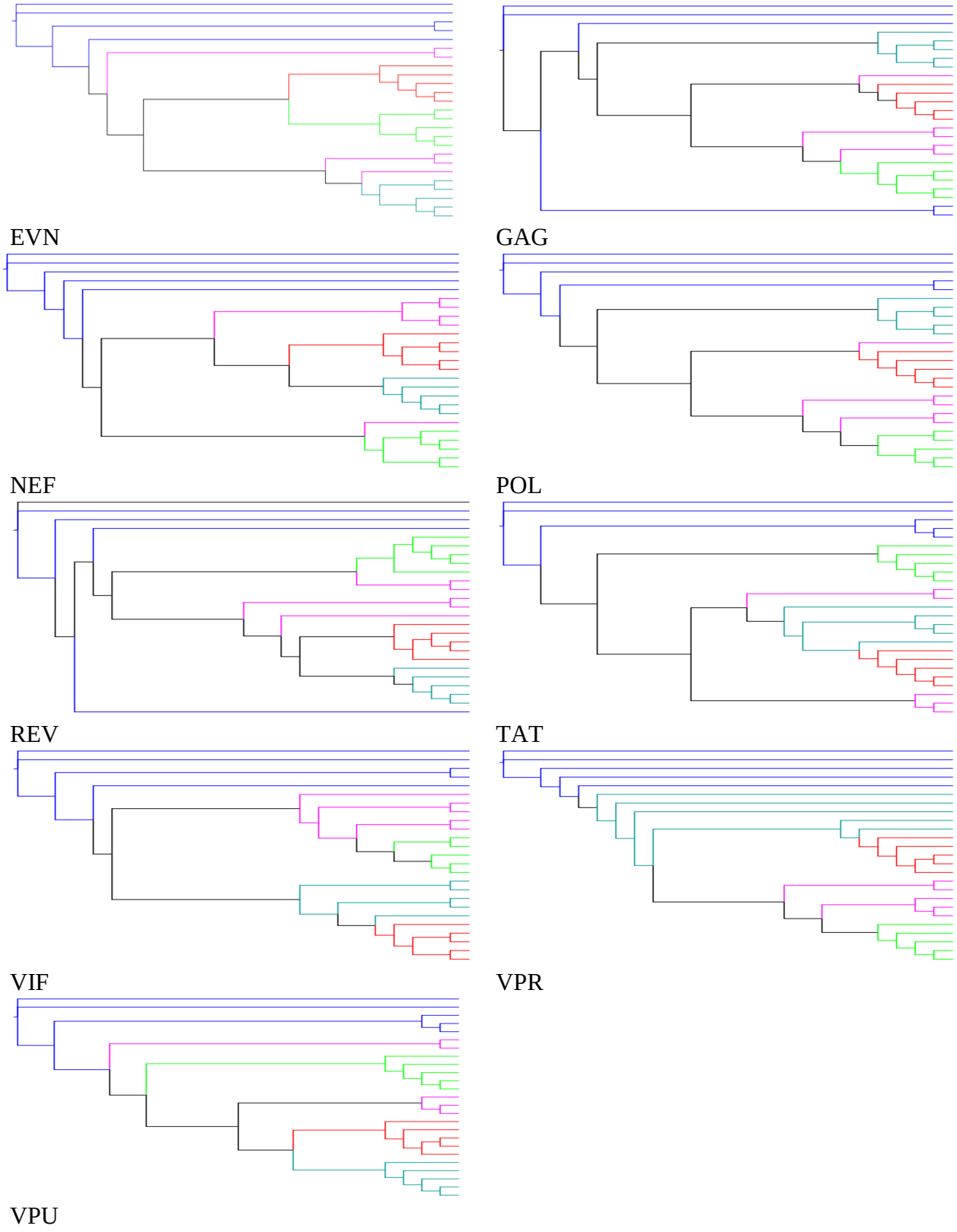


Figure D.3: Gene Trees for Batch 3. Individual Gene Classification Method

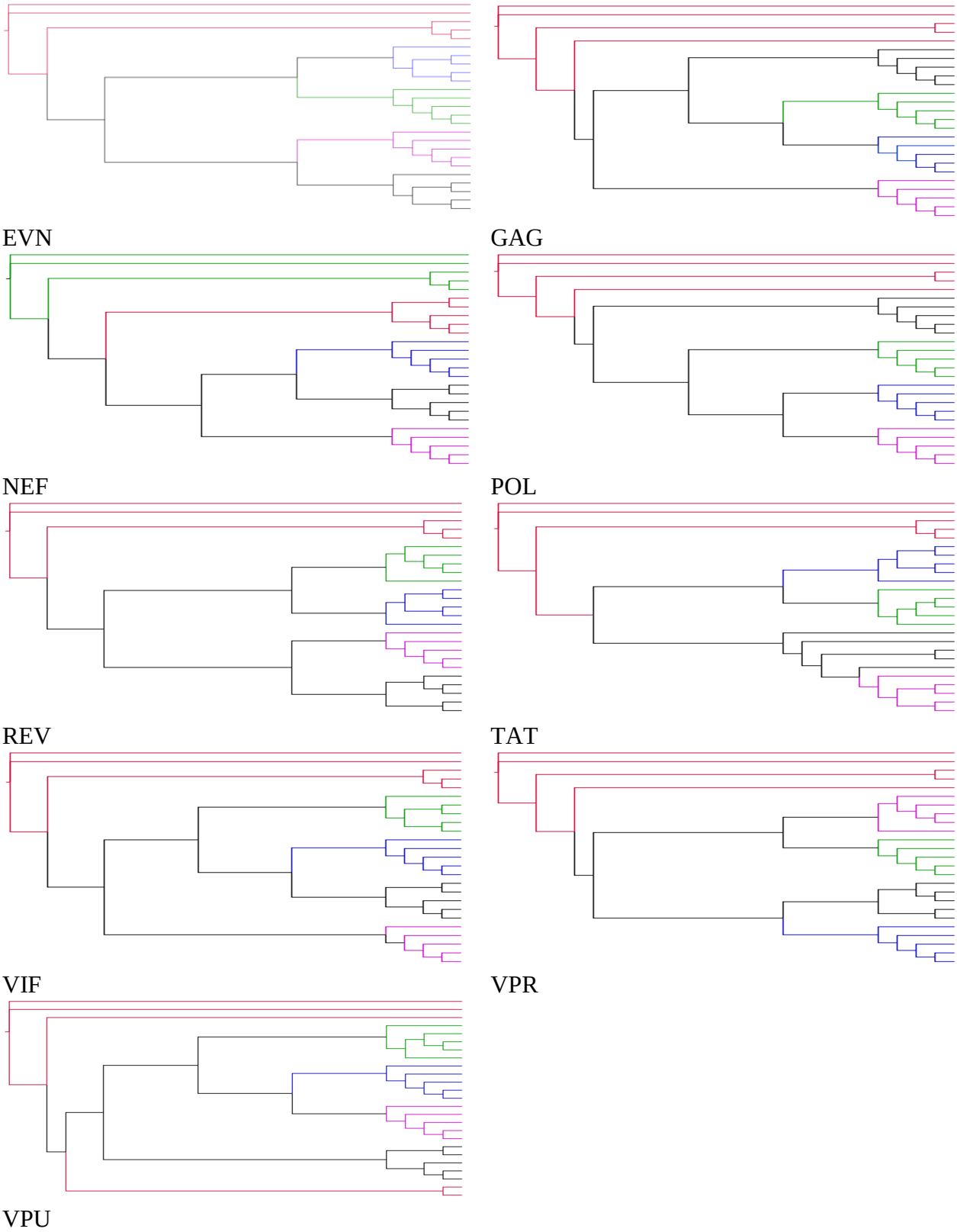


Figure D.4: Gene Trees for Batch 4. Individual Gene Classification Method

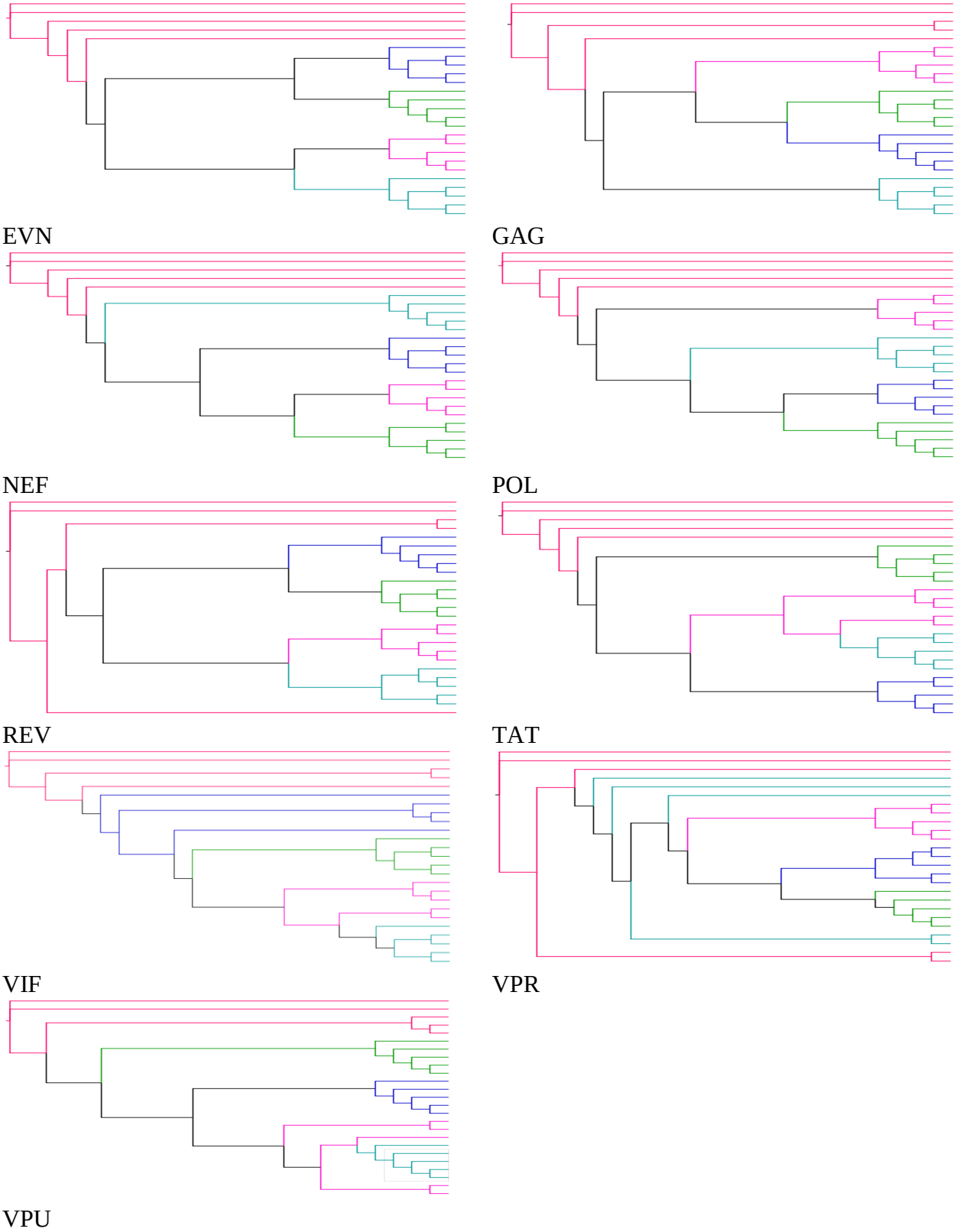


Figure D.5: Gene Trees for Batch 5. Individual Gene Classification Method

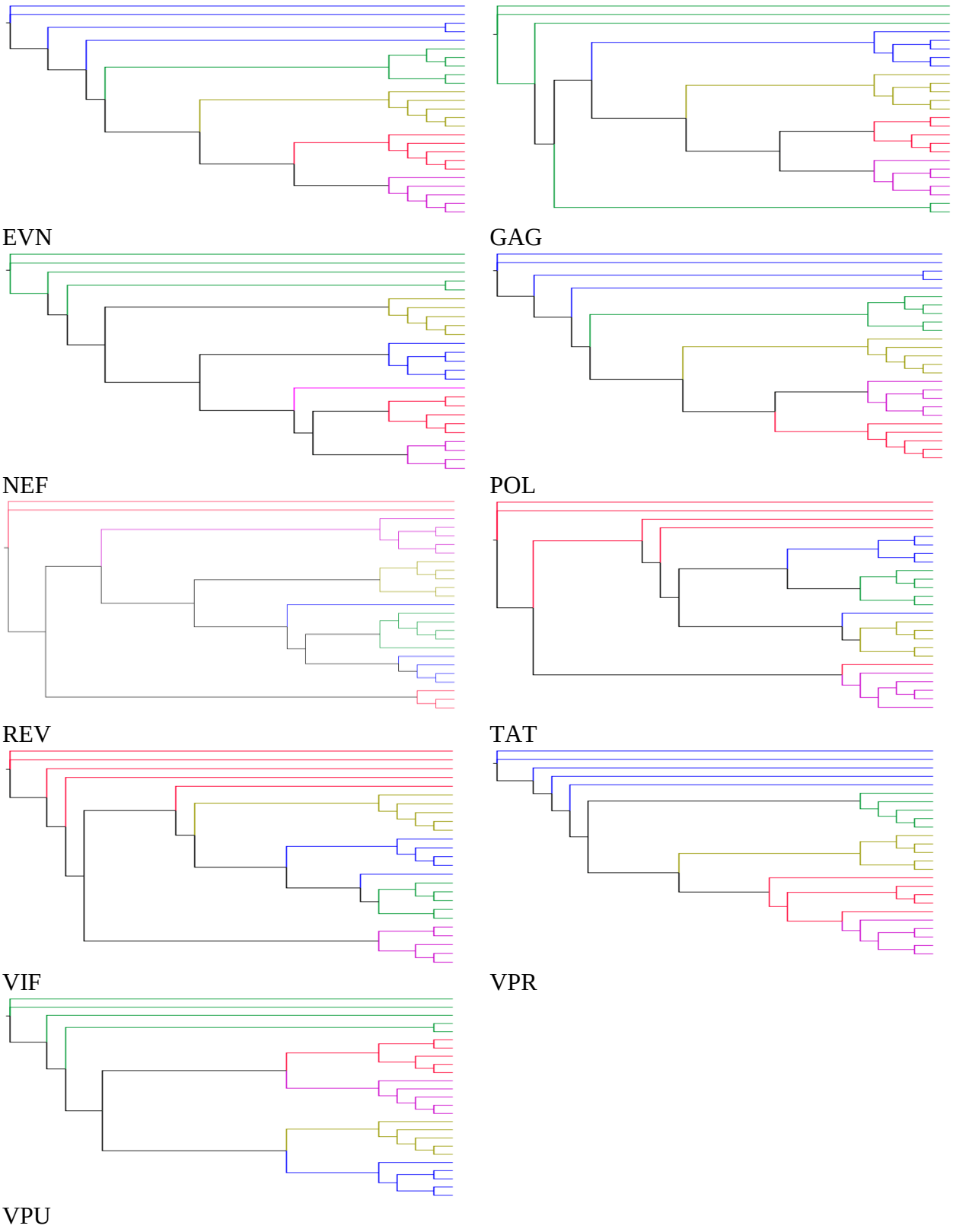
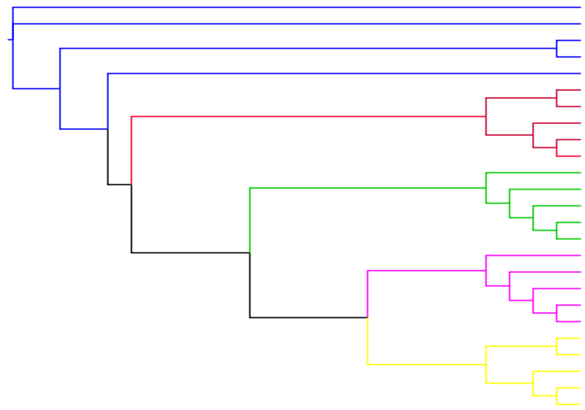
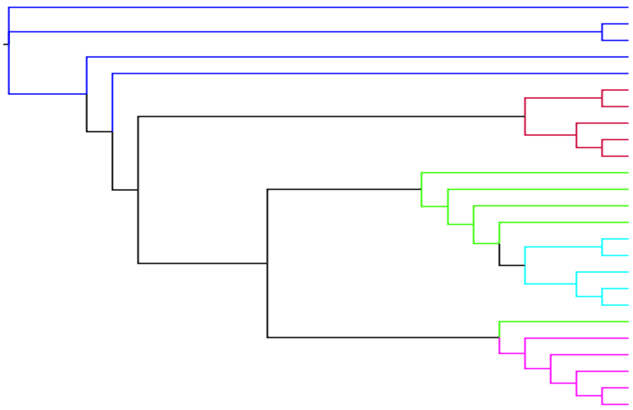


Figure D.6: Gene Trees for Batch 6. Individual Gene Classification Method

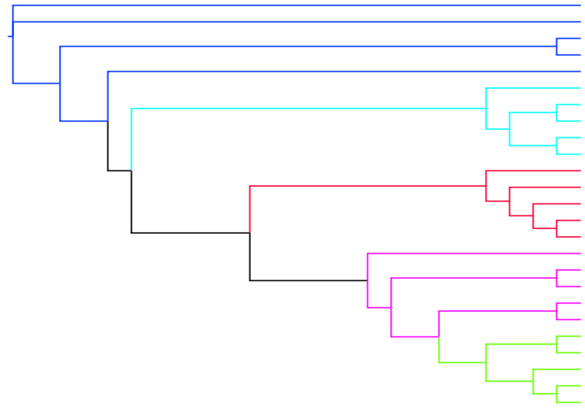
APPENDIX E: GENE TREES WHOLE GENOME



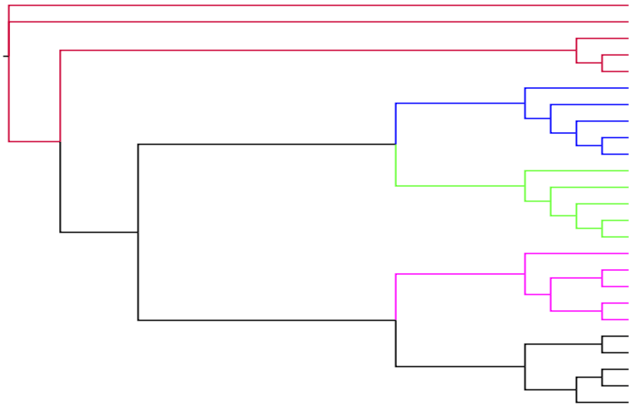
BATCH 1



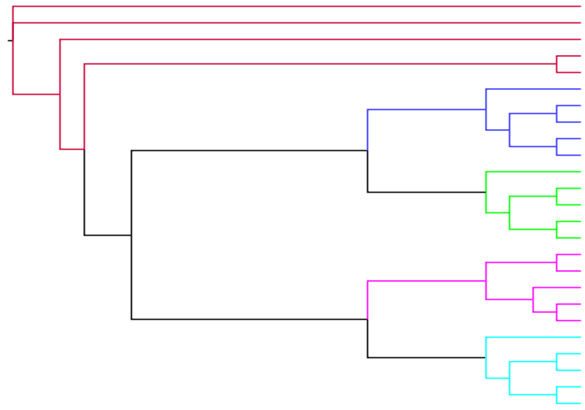
BATCH 2



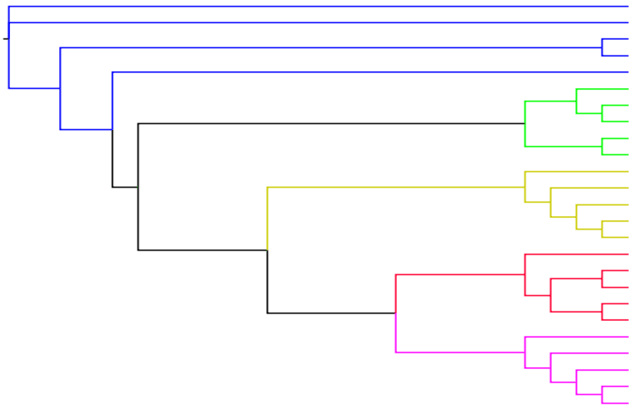
BATCH 3



BATCH 4



BATCH 5



BATCH 6

Figure E.1: Complete Genome Gene Trees for Batches 1-6. Concatenation Method

APPENDIX F: CONSENSUS

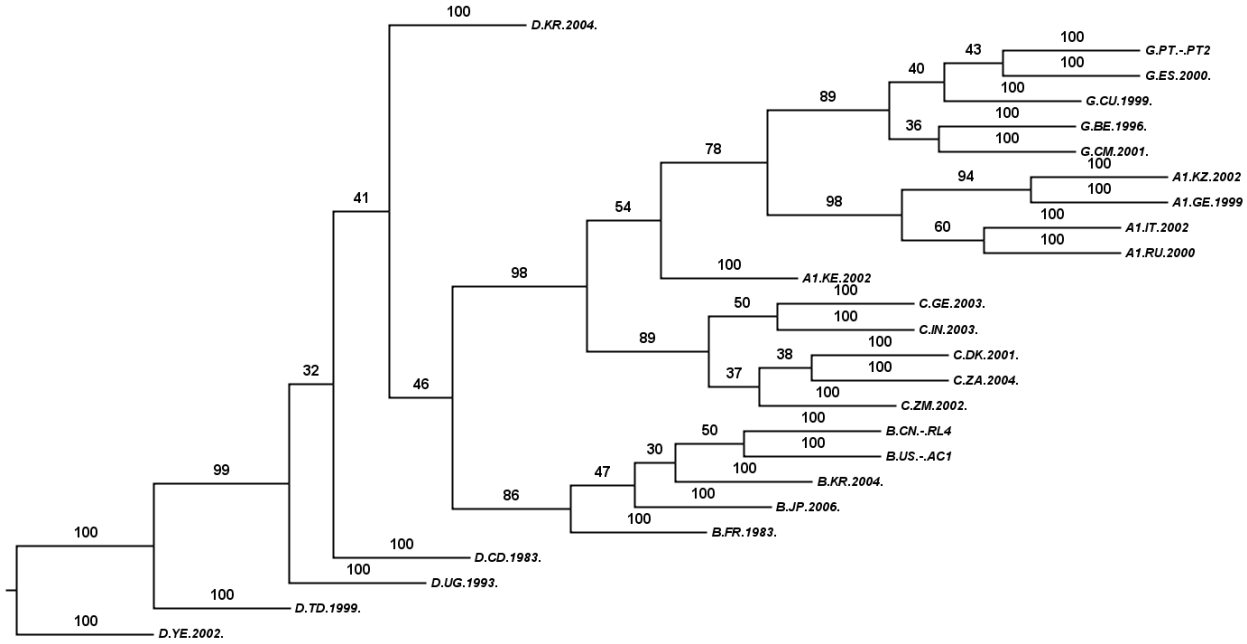


Figure F.1: Consensus Tree TAT