Studying Genetic Diversity and Evolutionary Pattern in Human Immunodeficiency Virus

Utilizing sequencing data and machine learning

Master's thesis in Engineering Mathematics and Computational science

Laleh Varghaei

DEPARTMENT OF MATHEMATICAL SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Studying Genetic Diversity and Evolutionary Pattern in Human Immunodeficiency Virus

Utilizing sequencing data and machine learning

LALEH VARGHAEI

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mathematical Sciences
Division of Applied Mathematics
Chalmers University of Technology
Gothenburg, Sweden 2024

Studying Genetic Diversity and Evolutionary Pattern in Human Immunodeficiency Virus
Utilizing sequencing data and machine learning
LALEH VARGHAEI

Supervisor: Erik Lorén, Bioinformatician at 1928 Diagnostics
Examiner: Erik Kristiansson, Full Professor in Applied Mathematics and Statistics

Master's Thesis 2024
Department of Mathematical Sciences
Division of Applied Mathematics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Studying Genetic Diversity and Evolutionary Pattern in Human Immunodeficiency Virus
Utilizing sequencing data and machine learning
LALEH VARGHAEI
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

The Acquired Immunodeficiency Syndrome (AIDS) pandemic has affected millions of people worldwide and posed a threat to global health. Since the discovery of the Human Immunodeficiency Virus (HIV) as the cause of the AIDS pandemic, numerous studies have been conducted on this virus, and many attempts have been made to develop an effective treatment or vaccine. HIV mutates very often, and it has many subtypes and variants, which makes developing an effective treatment challenging. Therefore, it is important to identify mutations that can lead to drug resistance as well as to identify the subtypes. Studying the evolutionary patterns of HIV is also crucial to understand where this pathogen comes from and what we can expect from it in the future. To identify Drug Resistant Mutations (DRMs), various subtypes, and conduct phylogenetic analysis of sequencing data, various bioinformatic tools and machine learning methods were employed. A pipeline was constructed by combining different bioinformatic software, which was capable of identifying low-frequency DRMs. For identifying different HIV subtypes and studying phylogenetic and evolutionary patterns, both bioinformatic tools and supervised machine learning methods were employed. Each of the two approaches applied succeeded in identifying subtypes and studying phylogenetic relationships, but the feature selection techniques in machine learning used for discovering evolutionary patterns had some limitations. The abundance of sequencing data enables the use of various approaches, such as machine learning, for studying viral genomes. This approach allows for a better understanding of the pathogen and can suggest appropriate solutions for combating it.

Keywords: Acquired Immunodeficiency Syndrome, Human Immunodeficiency Virus, Drug Resistant Mutation, Subtype, Bioinformatics, Machine learning, Sequencing

Acknowledgements

I would like to thank my supervisor, Erik Lorén, for his invaluable guidance and support throughout this project. I am equally thankful to my examiner, Erik Kristiansson, whose insights and constructive criticism have greatly enriched this work. My appreciation also goes to Iris Gold Rodal for her willingness to help, her time spent reading this report, and her valuable feedback. I would like to thank my opponent, Pernilla Huynh, for her thoughtful feedback and suggestions for improving the report. Lastly, I would like to express my gratitude to everyone at 1928 Diagnostics for their support and assistance in various forms, which made completing this project easier.

Laleh Varghaei, Gothenburg, April 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AIDS	Acquired Immunodeficiency Syndrome
CRF	Circulating Recombinant Form
CBOW	Continuous Bag-Of-Words
DNA	Deoxyribonucleic Acid
DRM	Drug Resistant Mutation
EBI	European Bioinformatics Institute
HCV	Hepatitis C Virus
HIV	Human Immunodeficiency Virus
INSTI	Integrase Strand Transfer Inhibitor
IDF	Inverse Document Frequency
KNN	K-Nearest Neighbors
LASSO	Least Absolute Shrinkage and Selection Operator
LDA	Linear Discriminant Analysis
MLE	Maximum Likelihood Estimation
MEGA	Molecular Evolutionary Genetics Analysis
NB	Naïve Bayes
NCBI	National Center for Biotechnology Information
NJ	Neighbor-Joining
NGS	Next Generation Sequencing
OTU	Operational Taxonomic Unit
PBMC	Peripheral Blood Mononuclear Cell
PCR	Polymerase Chain Reaction
PCA	Principal Component Analysis
pdf	probability density function
QDA	Quadratic Discriminant Analysis
RF	Random Forest
SHIV	Simian Human Immunodeficiency Virus
SIV	Simian Immunodeficiency Virus
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
UDS	Ultra Deep Sequencing
WHO	World Health Organization
URF	Unique Recombinant Form

Contents

List of Figures

List of Tables

1

Introduction

The advent of the Covid pandemic in 2020 made many people around the world aware of the nature of viruses and how mutations in viral genomes can give rise to new resistant variants that can lead to treatment and vaccine failure. Identifying mutations, especially low-frequency mutations in the viral genome, as well as identifying virus subtypes are crucial to combating pathogenic viruses. Moreover, it is important to study the evolutionary patterns and understand how the virus evolves over time. This approach helps us to understand which parts of the viral genome are more prone to mutation and which parts should be targeted in the treatments.

Genome sequencing is a crucial first step in virology but older techniques like Sanger sequencing has its limitation and can fail to identify low-frequency mutations, so now Next-Generation Sequencing (NGS) techniques are set to replace the old methods [1]. Beyond the capability of NGS to identify minor variants in viral samples, it is more cost-effective than Sanger sequencing, making it a suitable technique for use, especially in low-income countries.

The abundance of sequencing data makes it easier for researchers to study Deoxyribonucleic Acid (DNA). One approach to analyzing big and complex genomic data is using machine learning algorithms. Machine learning algorithms can be trained to predict the outcomes of data and they can recognize patterns in the dataset, such as identifying significant disease biomarkers in gene expression data [2].

Considering the advantages of NGS over traditional methods such as Sanger sequencing, the primary goal of this project is to study genetic diversity and evolutionary patterns in viruses using NGS data and machine learning methods. The virus to be studied in this project is Human Immunodeficiency Virus (HIV). Due to its significant impact on global health, HIV is one of the most thoroughly researched viruses in the world, with abundant information and data available in online databases. Although the main focus of this project is HIV, some of the developed methods were reapplied to other viruses such as the Hepatitis C Virus (HCV) to illustrate how these methods can be expanded to other viruses.

1.1 Aim and Scope

This project builds upon the results and findings of the previously completed project in the course "Individual Project in Mathematics and Mathematical Statistics" (MVE405, Chalmers University of Technology) and can be considered its continuation. In the previous project, virus genotyping was conducted with the aim of identifying potential resistance markers and subtypes. This goal is important for suggesting effective treatments for patients infected with HIV and for developing an effective vaccine. A pipeline was created to identify Drug Resistant Mutations (DRMs) in HIV, and some machine learning algorithms were used for subtyping. As a continuation of the previous project, the main goals of this project are as follows:

1. Improve the pipelines developed in the previous project, aiming at the identification of DRMs and subtyping.

2. Identify low-frequency mutations in viral samples.

3. Conduct phylogenetic analysis.

4. Investigate whether the developed pipelines and methods for HIV can be generalized for other viruses such as HCV.

This project was conducted at the Swedish company 1928 Diagnostics. 1928 Diagnostics is a bioinformatics company primarily focused on diagnosing and combating antibiotic resistance. Although initially concentrating on antibiotic resistance, the company is now expanding its expertise and techniques to include the study of viruses and fungi.

The project does not involve any laboratory experiments and is computationally based. The data used for all the steps, for both HIV and HCV, are obtained from publicly available online databases, such as the European Bioinformatics Institute (EBI), the National Center for Biotechnology Information (NCBI), and the Los Alamos National Laboratory.

2

Theory

In this chapter, the background theory on HIV, HCV, and the methodology used is provided.

2.1 Human Immunodeficiency Virus

HIV is a Ribonucleic Acid (RNA) virus that attacks a type of white blood cell in the immune system called CD4+ helper T cells. By suppressing the body's immune system, this virus can lead to Acquired Immunodeficiency Syndrome (AIDS), wherein the body becomes very weak and vulnerable to opportunistic infections and diseases. HIV is transmitted sexually, through blood, and from mother to child [3].

2.1.1 HIV origin

Since the emergence of HIV-1 in 1981, many have questioned the origin of this deadly virus. The first scientific theory about the origin of HIV-1 emerged in 1986, following the discovery of HIV-2 in West Africa. HIV-2, which could also lead to AIDS, was found to be distantly related to HIV-1. Scientists discovered that HIV-2 is genetically similar to a simian virus that causes immunodeficiency in captive macaques. Following this discovery, various Simian Immunodeficiency Viruses (SIV) were identified in different primates [4].

The findings showed that HIV-1 is genetically similar to a SIV that infects chimpanzees, and HIV-2 is genetically related to a SIV found in Sooty mangabey. Relating both types of HIV to a specific simian species is difficult due to the lack of sequencing data in different simian species. SIV in chimpanzees is thought to be the origin of HIV-1, but due to the lack of data, this theory remained somewhat unclear. After the discovery of SIV in gorillas, another candidate for the origin of HIV-1 has been found. Nowadays, it is thought that HIV-1 originates from either chimpanzees or gorillas. The sequencing data and research show that the origin of HIV-2, which is genetically distinct from HIV-1, is SIV, which infects Sooty mangabey. The genetic similarity between HIV and SIV indicates that both types of HIV are the result of cross-species transmission. This transmission could occur through contact with the blood of an animal during hunting or by consuming the infected meat of primates[4].

2.1.2 HIV types and subtypes

HIV has two main types: HIV-1 and HIV-2. The average genome size of HIV-1 is between 9200 and 9600 nucleotides, while that of HIV-2 is 9800 nucleotides [5]. Both HIV types have nine genes [6]. HIV-2 is not as easily transmitted as HIV-1, and it takes a longer time to lead to AIDS. Additionally, HIV-2 is more restricted to Western African countries and has not spread globally like HIV-1 [4]. The genetic similarity between HIV-1 and HIV-2 is only 55% [7].

Both HIV types have a high replication rate and are prone to errors and mutations when converting their RNA to DNA [4]. This leads to the existence of many subtypes and sub-subtypes within each corresponding type of HIV. HIV-1 is categorized into four main groups: M, N, O, and P. Among these, Group M is the most commonly transmitted worldwide and is further classified into nine subtypes: A, B, C, D, F, G, H, J, and K. Furthermore, each of these subtypes are divided into numerous sub-subtypes [8]. Genetic differences ranging from 25% to 35% can be observed within HIV-1 subtypes [9].

HIV-2 is not as transmissible as HIV-1, and not all cases of HIV-2 lead to AIDS. Also, transmission from mother to child does not occur with this type. This type is more restricted to West Africa and has not spread to other regions. HIV-2 has eight main subtypes, although only subtypes A and B have been observed in many patients. The other subtypes have been seen in only one individual and have not spread further. Over time, HIV-2 has been increasingly replaced by HIV-1 in West Africa[4]. The two main types and different subtypes of HIV belonging to major types is shown in Figure 2.1.

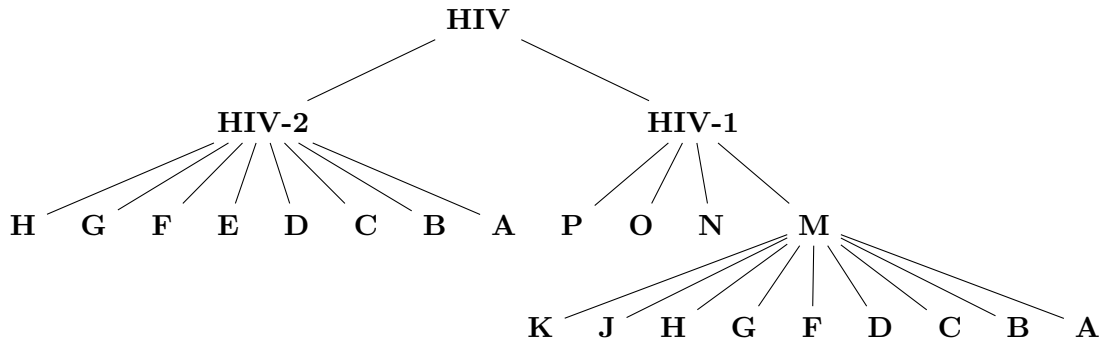

Figure 2.1: Classification tree of HIV, showing the major types and subtypes.

2.1.3 Circulating Recombinant Forms

Superinfection occurs when a person already infected with a strain of HIV from a certain subtype becomes reinfected with another strain of HIV from a different subtype. Approximately 10% of HIV infections worldwide are caused by Circulating Recombinant Forms (CRFs). HIV is diploid, meaning it carries two strands of RNA genome. When a cell in the body is infected with two different subtypes of HIV, the RNA from these different subtypes can accidentally mix, creating a new virion. This new virion, with the combined RNA, can infect another cell, and the reverse

transcriptase enzyme converts the mixed recipe of the two different RNAs into brand-new DNA. In this way, a new variant is produced, which presents both a threat and a challenge in combating the virus. In CRFs, also known as mosaic viruses, the combination of two different variants can occur in any part of the genome and in any nine gene's of HIV. This random combination can enhance the virus's adaptation to the host cell and help escaping from the immune response [10].

A second scenario can also occur, where the RNA from two different strains within the host's body can mix and combine, producing a new strain known as Unique Recombinant Form (URF). The key difference between CRFs and URFs is that CRFs can spread to other people, while URFs remain in the initial host and do not spread further. This could be due to randomness or a low capacity for URF replication that cannot compete with other dominant strains [10].

One of the dominant CRFs is CRF02_AG. According to Figure 2.3, CRF02_AG has become a dominant variant in a large part of Africa. Figure 2.2 shows the genetic structure of CRF02_AG, which results from the combination of subtype A and subtype G.

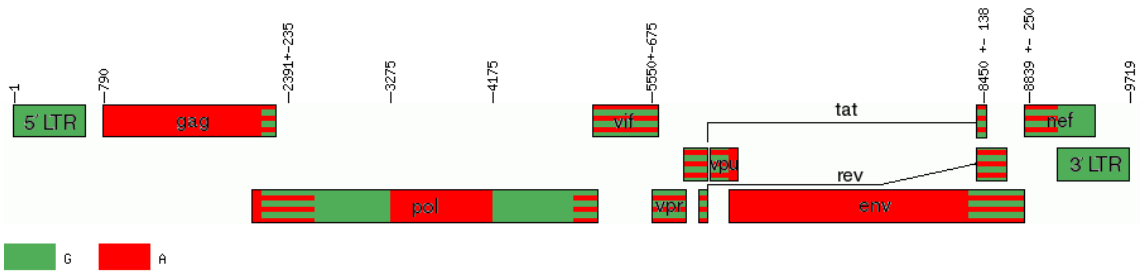

Figure 2.2: The genetic composition of CRF02_AG made by combination of subtype A and subtype G in HIV-1. The Figure is from Los Alamos National Laboratory databese [11].

More than 160 subtypes and CRFs are registered in the HIV database of the Los Alamos National Laboratory [12]. Figure 2.3 shows the dominant HIV-1 subtypes in each country according to the Los Alamos National Laboratory HIV database. This map does not represent the actual dominant HIV-1 subtypes in some countries, especially developing countries, due to the lack of sufficient sequencing data. However, Figure 2.3 demonstrates the diversity of HIV-1 around the world.

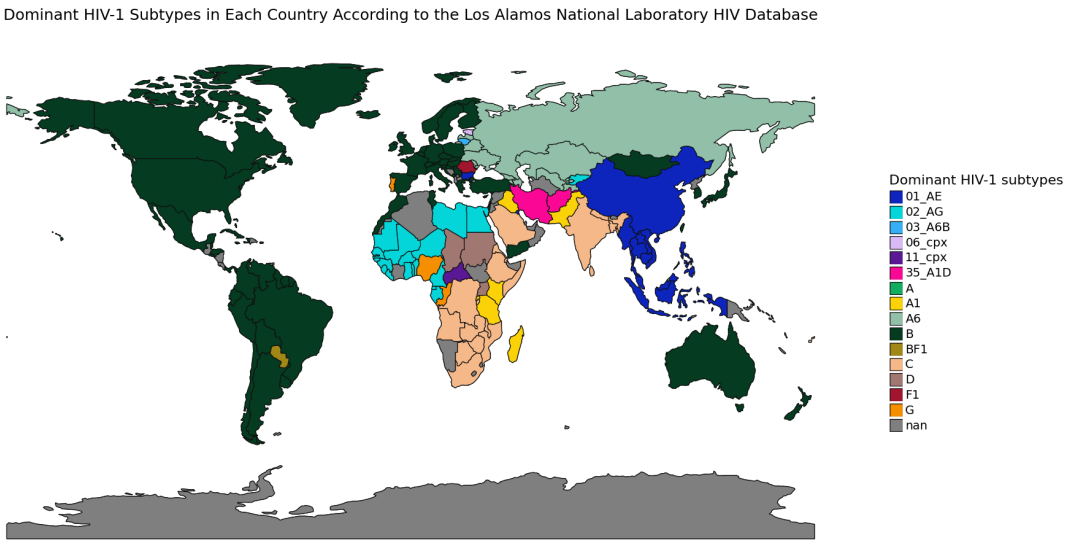

Figure 2.3: Dominant HIV-1 subtypes in each country according to the Los Alamos National Laboratory HIV database.

2.2 Hepatitis C Virus

HCV is a RNA virus from the family Flaviviridae which was first isolated in 1989. HCV by targeting a kind of liver cells, called hepatocytes, can lead to severe liver damage, including cirrhosis and cancer [13]. HCV does not only target hepatocytes, but recent studies also show that it significantly affects the function of Peripheral Blood Mononuclear Cells (PBMC) [14]. The transmission of HCV is similar to that of HIV, it can occur through sexual contact, from mother to child, and through infected blood. According to World Health Organization (WHO) 50 million people are infected currently by chronic HCV worldwide and about 242000 people died from HCV in 2022 [15]. Although the rate of HCV infection has decreased in recent years, it is still considered one of the deadliest viruses in the world and poses a threat to global health.

The viral RNA of HCV mutates quite often due to high replication and lack of error proofreading by the viral RNA polymerase [13]. According to a study conducted in 2019, HCV has 8 main genotypes and 86 subtypes [16]. All these genotypes are divided to several subtypes. The existence of many subtypes makes developing a functional vaccine that can provide protection against all genotypes and subtypes of HCV challenging. Fortunately, numerous effective medications and treatments have been developed to control the disease and prevent its progression to a chronic condition. Some of these antiviral treatments target the HCV virus by inhibiting its replication or preventing the virus from entering the host cells, while others focus on boosting the body's immune response. As previously mentioned, HCV affects not only the liver cells but also manipulates the function of the immune system cells, preventing them from eliminating the infected cells. Some medications

have been developed to stimulate the immune system to attack cells infected by HCV [17]. Similar to HIV, HCV frequently mutates and can develop resistance against some treatments. Moreover, each genotype and subtype has its own genetic characteristics and composition, making identification crucial before suggesting the appropriate treatment. Therefore, identifying DRMs and subtyping HCV plays a very important role in combating this virus.

2.3 Converting Sequences to Numerical Values

In order to use machine learning for the analysis of viral genomes, it is necessary to convert the sequences into numerical values, that is, to represent each sample by some features which are numbers. Two methods are used in this project to achieve this goal and are presented in this section.

2.3.1 Term Frequency-Inverse Document Frequency

Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical method in natural language processing. It calculates how often a term appears in a text while also considering the importance of that term. For example, terms such as "the", "a", and "and" may appear very often in a text but do not carry much information about the text as a whole. In this way, TF-IDF, by calculating the importance of each term, prevents bias towards any specific term [18].

Term Frequency (TF) represents how frequently a term appears in a text compared with the total number of the all terms:

$$\mathrm{TF}(t,d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in the document}} \tag{2.1}$$

Inverse Document Frequency (IDF) measures the importance of each term in a collection of documents by calculating the logarithm of the ratio between the total number of documents and the number of documents containing the term, in this way it assigns higher weights to terms that occur less frequently across the corpus:

$$\mathrm{IDF}(t,D) = \log\left(\frac{\text{Total number of documents in the corpus } D}{\text{Number of documents with term } t \text{ in it}}\right) \tag{2.2}$$

The final formula for TF-IDF is as following:

$$\text{TF-IDF}(t,d,D) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t,D) \tag{2.3}$$

Where:

- t is the term.

- d is the document.
- D is the corpus (the collection of all documents) [18].

2.3.2 Word to Vector

Word to vector or Word2Vec is a two-layer neural network used in natural language processing to represent terms or words with vectors, a common method in machine learning for word embedding. This algorithm is distinctive because it preserves the semantic meaning of words or terms. Word2Vec captures not only the semantics of words or terms but also the relationships between a term and its surrounding terms. Each word is represented by a multidimensional vector that contains the word's semantics and can preserve various linguistic patterns. Words with similar semantics end up in close vector positions. For example, the word "king" means a male ruler, "man" carries the concept of maleness, and "woman" the concept of femaleness. By subtracting the "male" vector from the "king" vector and adding the "female" vector, the word "queen" is obtained [19] [20]:

$$\mathbf{v}_{\text{king}} - \mathbf{v}_{\text{man}} + \mathbf{v}_{\text{woman}} \approx \mathbf{v}_{\text{queen}} \tag{2.4}$$

Word2Vec represents each word with vectors using two methods:

1. **Continuous bag-of-words (CBOW)**: In this method, the algorithm predicts a word based on the context, that is, the surrounding words of the target word.

2. **Skip-gram**: This method works in the opposite way of CBOW, meaning it predicts the surrounding words of a target word based on the target word itself [19] [20].

2.4 Phylogenetic Analysis

Like other biological organisms, viruses evolve over time, and identifying the ancestor of a specific virus aids in combating it. Furthermore, understanding the relationship of a particular pathogen with other pathogens can provide information about its functions. In infection control, identifying the relatedness of two samples can help in identifying their source. Phylogeny involves the study of evolutionary relationships between different species, illustrating how various organisms are related through the process of evolution [21].

Another branch of phylogenetic analysis involves predicting evolutionary patterns. Due to viruses short generation times and high mutation rates, it is expected that new virus variants, potentially resistant to current treatments, will emerge in the future [22]. Thus, predictive evolutionary studies are important for identifying which parts of the genome are more likely to evolve or change.

To conduct phylogenetic analysis of HIV-1 a software called Molecular Evolutionary Genetics Analysis (MEGA) is used [23]. To conduct predictive evolutionary analysis, two feature selection techniques in machine learning are employed, which will be explained in this section.

2.4.1 Molecular Evolutionary Genetics Analysis

MEGA was first released in 1993 and has been incorporated many new and additional methods to perform accurate evolutionary analyses of different species. MEGA offers several methods for constructing phylogenetic trees [23]:

- Maximum Parsimony
- Neighbor-Joining (NJ)
- Maximum Likelihood Estimation (MLE)
- UPGMA
- Minimum-Evolution

In this part, the MLE and NJ methods that are used in this project are presented.

2.4.1.1 Maximum Likelihood Estimation

MLE is a statistical method that helps estimate the parameters of a model. The estimation of the parameters is done by finding the best value for the parameters in a way that maximizes the probability of the observed data occurring. To facilitate the calculation of the joint probability distribution of all observed data in the sample, an assumption is made. The first assumption in MLE is the assumption of independence, meaning all the data points are independent of each other [24].

According to the first assumption, the observations $x_1, x_2, \ldots, x_n$ are independent and identically distributed. The likelihood function $L(\theta|x_1, x_2, \ldots, x_n)$, which is a function of the model's parameters given the observed data, is then the product of the probabilities of observing each data point individually, or the product of the probability density functions (pdfs) for each data point [24] [25]:

$$L(\theta|x_1, x_2, \ldots, x_n) = f(x_1|\theta) \times f(x_2|\theta) \times \ldots \times f(x_n|\theta) \tag{2.5}$$

The method for maximizing the maximum likelihood function involves differentiating it and setting it to zero. This process can be simplified by taking the natural logarithm of the function [24] [25]:

$$\ell(\theta|x_1, x_2, \ldots, x_n) = \ln L(\theta|x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \ln f(x_i|\theta) \tag{2.6}$$

$$\frac{\partial \ell(\theta|x_1, x_2, \ldots, x_n)}{\partial \theta} = 0 \tag{2.7}$$

2.4.1.2 Neighbor-Joining

NJ is a common method used in bioinformatics for creating a phylogenetic tree. This distance-based method involves constructing a matrix that contains the pairwise distances of all Operational Taxonomic Units (OTUs). The term neighbor is defined as two OTUs that are linked by one interior node in a tree and splits into two branches at each point. The process begins with a star-like tree to which all the OTUs are connected, illustrated in Figure 2.4. At each step, it selects the two closest neighbors and joins them at a new node, like in Figure 2.5. The distances between this new node and the remaining OTUs are then recalculated, and the process repeats until all OTUs have been joined into a single phylogenetic tree [26].

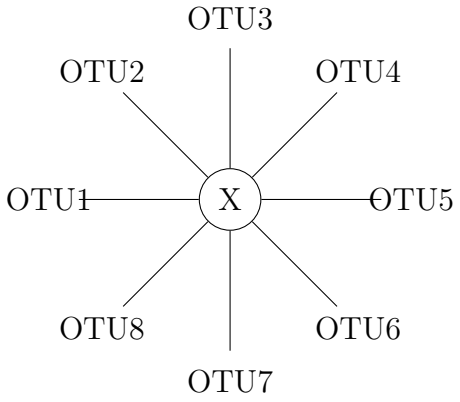

Figure 2.4: In the first step of NJ all OTUs are connected to a common and central node [26].

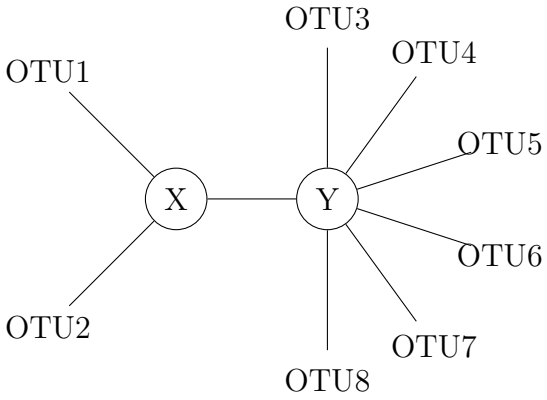

Figure 2.5: In the next step, OTUs that are more closely related to each other are grouped together under a new node. OTU1 and OTU2 are referred to as neighbors [26].

2.4.2 Feature Selection Methods

Feature selection is one of the important concepts in machine learning. Variables are very important when working with big data because they provide crucial information about the data points, but this is not always the case. Sometimes, some variables are simply noise and their existence is irrelevant. Noise or irrelevant variables not only increase the model's complexity but also can cause overfitting, meaning that the model captures these random noises as if they were meaningful patterns. Here, feature selection plays an important role. It reduces the input variables in the model by identifying and choosing only the most relevant features[27]. In this part, two feature selection techniques that are used in this project are presented.

2.4.2.1 Least Absolute Shrinkage and Selection Operator

Least Absolute Shrinkage and Selection Operator (LASSO) is a statistical model that aims to balance model simplicity with achieving good accuracy. The difference between LASSO and regular linear regression is that LASSO has the ability to shrink the variables. Lasso retains only the most important variables for prediction by pushing irrelevant variables toward zero. This is achieved through L1 regularization, which involves adding a penalty to the absolute size of the regression coefficients [28]. The formula for L1 regularization is as follows :

$$\alpha \sum_{j=1}^{p} |\beta_j| \tag{2.8}$$

Where:

- α is the regularization parameter.
- β_j are the coefficients.

The purpose of LASSO regression is to find the set of coefficients that results in the smallest possible value of the cost function, while simultaneously minimizing the L1 regularization parameter:

$$\text{Minimize: } \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \alpha \sum_{j=1}^{p} |\beta_j| \tag{2.9}$$

Where:

- y_i is the actual value of the response variable.
- x_{ij} is the value of the predictor variable.
- α is the regularization parameter.
- β_j are the coefficients [29].

2.4.2.2 Elastic Net

Elastic Net is a statistical model that combines both Ridge and LASSO regression. By merging these two methods and finding a balance between Ridge and LASSO regression, Elastic Net aims to keep the advantages of both methods while minimizing their weaknesses, in order to handle large datasets with high correlation [30]. Ridge regression or L2 regularization, unlike LASSO regression, does not shrink the coefficients of some variables to zero but instead reduces their magnitude more evenly. This approach ensures that all features remain part of the model but with minimized influence, allowing for more stable predictions when dealing with highly

correlated variables or when the number of features are more than the number of observations [31][32].

3

Methods

This thesis is based on the results and findings of the previously completed project in the course "Individual Project in Mathematics and Mathematical Statistics". In the previous project, a pipeline was developed to identify DRMs in HIV-1. Subsequently, both supervised and unsupervised machine learning methods were utilized for HIV-1 subtyping. Continuing from the previous project, we first aim to refine the two developed models for DRM and subtyping pipelines. Later, we tailored the developed pipeline specifically for identifying low-frequency mutations. The project will proceed with a phylogenetic analysis of HIV-1 to understand its origins and evolutionary patterns. Furthermore, the developed models were tested on HCV to determine whether the methods are specialized or can be considered a general pipeline for different viruses. All the code is written in Python and it is available in Bitbucket of 1928 Diagnostics [33].

3.1 Refining DRM and subtyping pipelines

In the previous project a pipeline was made for identifying DRMs in HIV-1. The different steps of the pipeline work as follows:

1. Take inputs from Illumina sequencing in FASTQ format.
2. Check the quality of reads in FASTQ files with **FastQC** (v.0.11.9) [34].
3. Perform sequence trimming using **Fastp** (v.0.20.1) [35] .
4. Align sequences with **Bowtie2** (v.2.4.4) [36] .
5. Use **Freebayes** (v.1.3.6) [37] for variant calling.
6. Generate a consensus sequence using **Bcftools** (v.1.13) [38].
7. Use **Sierralocal** (v.0.2.1) [39] to identify DRMs.

A patient's sample, sourced from EBI [40], was tested using the developed pipeline and simultaneously run through two other existing HIV pipelines, Hivmmer [41] and Quasiflow [42], as a validation step. The DRMs identified by the three pipelines were plotted as a bar plot and compared. Our pipeline missed some significant

DRMs that were identified by the other two pipelines, which is why the different steps of the pipeline were further analyzed. Other software, such as BWA (v.0.7.17) [43] for alignment and LoFreq (v.2.1.2) [44] for variant calling, were used. Later, the reference genome used in all the steps was analyzed, especially the software like Sierralocal (v.0.2.1), which automatically uses its own reference genome.

After testing various software for the alignment and variant calling steps, it was decided that the current tools, Bowtie2 (v.2.4.4) and Freebayes (v.1.3.6), are functioning well and will be retained in the pipeline. Additional parameters were added to the pipeline to increase its sensitivity. The additional parameters added to the variant calling steps are as follows:

- **Minimum alternate count:** This parameter controls the minimum number of observations required to consider an alternate allele as a variant and it is set to one in the pipeline.
- **Minimum alternate fraction:** This parameter determines the minimum fraction of reads necessary to consider an alternate allele as a variant. It is set to 0.009 in the pipeline, meaning that the variant must be supported by at least 0.9% of the total reads to be considered.
- **Minimum coverage:** This parameter sets a threshold for the minimum number of reads that must cover a position for it to be considered in variant calling. In the developed pipeline, a position must be covered by at least 100 reads to be evaluated for variant calling.
- **Minimum base quality:** It defines the minimum quality score that a nucleotide base must have to be considered in variant calling. The base quality is the same as the Phred quality score, which is encoded as ASCII characters in FASTQ files [45]. Setting a high base quality is crucial to distinguish between true variants and sequencing errors. The minimum base quality is set to 20 in the pipeline. The formula for base quality or Phred score is as follows:

$$Q = -10 \log_{10} P \tag{3.1}$$

  Where:

  - Q is Phred score or base quality.
  - P is the probability that the base call is incorrect [46].

- **Minimum mapping quality:** This specifies the minimum mapping quality a read must have to be considered in variant calling. It indicates the likelihood that the alignment is correct. Setting a high mapping quality is important to ensure the reliability of the variant calling process by filtering out reads that do not meet a certain threshold of mapping confidence. The minimum mapping quality is set to 20 in the pipeline. The formula for mapping quality

is similar to base quality and is as follows:

$$\text{Mapping Quality} = -10 \log_{10} P \tag{3.2}$$

Where:

- P is the probability that the mapping position is incorrect [47].

To perform subtyping, both supervised and unsupervised machine learning methods were utilized in the previous project. The supervised learning algorithms, such as classification, use the labels of the dataset to predict outcomes. Five classifiers were used for subtyping:

- Quadratic Discriminant Analysis (QDA)
- Linear Discriminant Analysis (LDA)
- Random Forest (RF)
- Naïve Bayes (NB)
- K-Nearest Neighbors (KNN)

The critical part of using these classifiers as a subtyping method was converting the sequences into numerical values. To achieve this goal, the sequences were divided into subsequences, called k-mers of a certain length. These subsequences or k-mers were used as features for each sample, and a value was given to each k-mer in each sample based on that k-mer's importance and abundance in each sequence with the help of TF-IDF. The optimal length of k-mers was decided after plotting the accuracy of each classifier versus the k-mer length. A dataset for subtyping HIV-1 was derived from the HIV database of the Los Alamos National Laboratory[12].

Although the result from converting k-mers to numerical values was satisfying, we aimed to test other methods. Another method tested was Word to Vector, or Word2Vec. After converting the sequences into k-mers, each k-mer was represented by a vector of a certain dimension. To make the data suitable for classification, we calculated the mean of all the vectors in each sequence. Averaging the vectors resulted in a single vector that represented the entire sequence. The optimal dimensions of the vectors and the optimal k-mer size were determined by plotting the accuracy obtained from the KNN classifier against the lengths of the k-mers and the dimensions of the vectors. In this approach, the sequence is no longer represented by k-mers as features, but by a single vector, with the features being the dimensions of that vector. After converting the sequences to numerical values the same classifiers were used for subtyping.

3.2 Identifying low-frequency mutations

The pipeline developed for identifying DRMs is used to identify minority variants. To identify these minority variants while preventing the identification of noise, some parameters were added to the variant calling step in the pipeline, which is done by Freebayes (v.1.3.6). Parameters such as minimum alternate fraction and minimum coverage, help to identify minority variants, and keeping the minimum base and mapping quality high, helps to prevent noise and mistakenly identifying errors as mutations.

To identify minority variants, we first generated synthetic data. This data was created in such a way that three sequences were produced. The first sequence is simply the reference genome. The second sequence contains a DRM I84V, and the third has DRM L90M. Combining all the three sequences, forward and reverse Illumina FASTQ files were created with the help of Biopython and InSilicoSeq (v.1.6.0) [48], a sequencing simulator. To introduce low-frequency mutations into our sample, an abundance file was created and given to InSilicoSeq (v.1.6.0) as input. The abundance of the reference sequence, without any mutation, is 98%, and the abundance of sequences with the introduced mutations is 1% each. After creating the FASTQ files, they were run in the pipeline.

After testing the pipeline using synthetic data, we aimed to evaluate the clinical impact of minority resistant variants by examining real data. The data used for this task belongs to the study "Prevalence and clinical impact of minority resistant variants in patients failing an integrase inhibitor-based regimen by ultra-deep sequencing" [1]. The data was collected between January 2014 and March 2017 at the Pitié-Salpêtrière, Saint-Antoine, and Bichat hospitals in Paris, France. They performed two sequencing methods, sanger sequencing and Ultra Deep Sequencing (UDS), Illumina, on plasma samples of 134 patients who failed Integrase Strand Transfer Inhibitors (INSTIs) treatment. The integrase gene was sequenced using Polymerase Chain Reaction (PCR), and results from both Sanger sequencing and Illumina were compared. The Illumina sequences were submitted to GenBank with the accession number SRP137063 [49]. These samples were downloaded and run through both our pipeline and Quasiflow as a validation step.

In the study, mutations were classified as majority variants when they made up more than 20% of the viral population and were identified by both Sanger sequencing and Illumina, and as minority variant when they made up less than 20% of the viral population and were only detected by Illumina [1]. In our project, we are working only with Illumina samples to identify low-frequency mutations. We define mutations as minor if they are present in less than 20% of the viral sequences in a sample. A mutation is considered to be a majority variant if it constitutes more than 20% of the viral sequences in a sample. To capture low-frequency mutations, we set the minimum alternate fraction parameter to 0.009. Also, to minimize the chance of mistaking noise for mutations, we set the minimum base quality and minimum mapping quality to 20.

3.3 Phylogenetic Analysis

In this part, the methods for phylogenetic analysis are presented. MEGA software (v.11) is used for building phylogenetic trees, and two feature selection methods in machine learning are used for studying the evolutionary patterns in HIV-1.

3.3.1 Phylogenetic Tree

To construct a phylogenetic tree and study the evolutionary relationships between viruses similar to HIV-1, a dataset was created from four viruses, HIV-1, HIV-2, Simian Human Immunodeficiency Virus (SHIV), SIV, along with their corresponding subtypes [12]. It should be mentioned that in the dataset, the data is not solely from NGS clones. In order to include all HIV-1 and HIV-2 subtypes and CRFs, we did not restrict ourselves to only NGS data, because some subtypes are very rare and NGS samples of these subtypes were not found. Moreover, not all SIV subtypes were available in the form of NGS data either. Many of these animals live in the wild, making it difficult to capture and sequence them. That is why both datasets contain data from other sequencing methods as well.

SHIV infects both humans and non-human primates, while SIV infects only non-human primates. It should be noted that not all HIV-1 subtypes are included in the dataset to prevent the tree from becoming too complicated. The data needs to be preprocessed before being inputted into MEGA (v.11). Since MEGA (v.11) accepts a multi-fasta file, all the sequences were merged into a multi-fasta file using Biopython. Then, the data containing the four viruses were input into MEGA (v.11) using the MLE method to construct a phylogenetic tree.

3.3.2 Predictive Evolutionary Analysis

To study the evolutionary pattern in HIV-1, we decided to focus on codons. A codon consists of three specific nucleotide that can code for a certain amino acid or decides the start or termination of the translation process [50]. Our aim was to investigate which codon is crucial for subtyping, that is, for identifying new subtypes. To achieve this goal, the data used for subtyping in the previous project [12] was utilized again. This time, the sequences were divided into k-mers of length three to represent the codons. Before dividing the sequences into k-mers, the sequences were filtered, and those containing any nucleotide other than Adenine (A), Cytosine (C), Thymine (T), and Guanine (G) were removed from the dataset. A, T, C, and G are the four main bases in the DNA. We remove the sequences that contain any other nucleotide than these four main bases in order to use only codons as features and to prevent any other features that could cause noise. From 1149 samples, 379 were removed, resulting in a dataset with 770 samples and 20 subtypes.

Two feature selection techniques are used for HIV-1 data to identify the most important and relevant codons for subtyping. By identifying these codons and, subsequently, the amino acids, one can determine which amino acid changes are crucial in generating new subtypes. Later, the positions of these relevant codons will be

determined in all sequences to see if any pattern appears and if they belong to a specific gene in HIV-1. After preprocessing, dividing the sequences into k-mers of length three, and converting the k-mers into numerical values using TF-IDF, two feature selection methods, LASSO and Elastic Net, were used to identify the most important codons.

Before applying LASSO on our data, α parameter needs to be tuned. α is the penalty term that decides the amount of shrinkage that will be implemented in the model. α adjusts the strength of the model, a higher α simplifies the model by pushing more coefficients to zero, and lower α allows more complexity by letting more coefficients be non-zero.

The optimal value of α is determined by first dividing the data into 80% for training and 20% for testing. Then, a 10-fold cross-validation is performed only on the training data, involving 1000 iterations to test different values of α with the goal of achieving the highest score. After identifying the optimal α, LASSO regression is applied to the test data, from which the most relevant variables are extracted. The computation of TF-IDF is applied first to the training data, and then the same transformation is applied to the test data to prevent data leakage.

The process of using Elastic Net as a feature selection technique is exactly the same as LASSO, with the difference that here we need to tune two parameters. Like LASSO regression, the optimal parameter α needs to be determined, and the L1 ratio also needs to be decided. The L1 ratio is a parameter that balances between LASSO and Ridge regression. If the value of the L1 ratio is closer to 1, it means that the regression model is more similar to LASSO. As the value of the L1 ratio approaches 0, the model becomes more similar to Ridge regression. It should be mentioned that the data were standardized before using LASSO and Elastic Net in order to make all the variables of the same scale.

3.3.3 Identifying New Variants

In order to identify new variants or subtypes, we aimed to test the developed methods to determine which one is more appealing and suitable for identifying new variants. First, a dataset was created containing nine subtypes and CRFs, CRF06_cpx, A1C, CRF01_AE, A1, B, CRF02_AG, D, C and CRF60_BC. Subtypes B and D, along with the CRF02_AG, each have two observations, while the rest have one observation each. All the samples are NGS clones obtained from Los Alamos National laboratory HIV database [12].

Five classifiers in the subtyping task were trained on 20 different subtypes and CRFs for identifying subtypes and CRFs in unseen data. The training data used for the subtyping task includes the subtypes and CRFs 01_AE, A1, B, 02_AG, D, and C, but it was not trained to recognize CRFs 06_cpx, A1C, and 60_BC. This data is given to the trained classifiers for prediction.

Our aim in this task is to test this data both with classifiers and with phyloge-

netic methods in MEGA (v.11). The same data is preprocessed with the help of Biopython, and a multi-fasta file is created. Then, a phylogenetic tree is built to observe the relationship between the different variants using the NJ method in MEGA (v.11).

3.4 HCV Subtyping

The subtyping method for HCV is exactly the same that was applied for HIV-1. The five classifiers QDA, LDA, RF, NB and KNN were employed for HCV subtyping. Unsupervised machine learning methods like clustering were not used for HCV, as supervised algorithms have proven to be very effective in predicting the subtypes.

The data for HCV were obtained from the Los Alamos National Laboratory [51]. The dataset includes subtypes 1a, 1b, 2a, and 3a. Subtype 1a has 1662 samples, subtype 1b includes 629 samples, subtype 2a contains 227 samples, and subtype 3a has 190 samples, resulting in a dataset with 2708 observations. Subtyping is performed using five classifiers, QDA, LDA, RF, KNN, and NB. The sequences were converted into k-mers of length eight. Then, the data was split into training and test sets through 10-fold cross-validation. Subsequently, the training data was transformed into numerical values using the TF-IDF method. To prevent data leakage, the test data was transformed into numerical values using the transformation applied to the training data. Finally, the training data was projected onto the first 25 principal components of the training data, and the test data was similarly projected onto these 25 principal components.

4

Results and Discussion

In this section the result from refining DRM and subtyping pipelines, identifying low-frequency mutations, phylogenetic analysis and applying the developed methods on HCV is presented and discussed.

4.1 Refining DRM and subtyping pipelines

Figure 4.1 shows the results of running the sample obtained from EBI [40] through our pipeline, as well as through two other existing HIV pipelines. The x-axis presents the DRMs identified by all the pipelines. As shown in Figure 4.1, our pipeline failed to identify some mutations that were detected by the other pipelines. It is unknown whether these mutations actually exist in the patient's sample, but there is a greater likelihood that mutations identified by both Hivmmer and Quasiflow are true positives.

In order to capture the missing mutations, different solutions were tested. Changing the software used for the aligning and variant calling steps from Bowtie2 (v.2.4.4) and Freebayes (v.1.3.6) to BWA (v.0.7.17) and LoFreq (v.2.1.2) did not help in improving the results or identifying the missed mutations.

Another effort to evaluate the performance of the pipeline was testing it with synthetic data. Because of having a complete control and information over the generated data, we could have a very clear expectation of the data output. This approach helps with determining if the pipeline is functioning well or not. The synthetic data was generated to test the pipeline for identifying low-frequency mutations. The data contains 98% reference viral sequences, 1% of a viral sequence containing the DRM I84V, and 1% DRM containing L90M.

After running the forward and reverse FASTQ files through the pipeline, we noticed that the pipeline is reporting DRM D232N in the integrase region, which was not present in the viral samples. This result was a warning that there is a bug and a problem in the pipeline. In order to solve it, the reference genome used in different steps of the pipeline was further analyzed. The reference genome used in all steps except for the last steps was the same. The last step of the pipeline, which uses Sierralocal (v.0.2.1) for identifying DRMs, uses a built-in reference genome. This reference genome was found and compared with the reference genome that we fed into the pipeline, in BLAST. The two reference genomes had a similarity of 99.31%. Our pipeline considered the 0.69% differences in these two reference genomes as a

variant, and in this way, a false positive was reported. By changing the reference genome and using the same reference across all the steps, this problem in the pipeline was solved.

To increase the sensitivity of the pipeline and find missed DRMs, some parameters were added to the variant calling step. The parameters, minimum alternate fraction, minimum alternate count, and minimum coverage, were changed respectively from 0.05, 2, and 0 to 0.009, 1, and 100. Additionally, minimum base quality and minimum mapping quality were raised from 0 and 1 to 20, respectively. After increasing the threshold in our pipeline to detect low-frequency mutations while avoiding noise capture, the sample was run again in our pipeline. The result of this second run is illustrated in Figure 4.2.

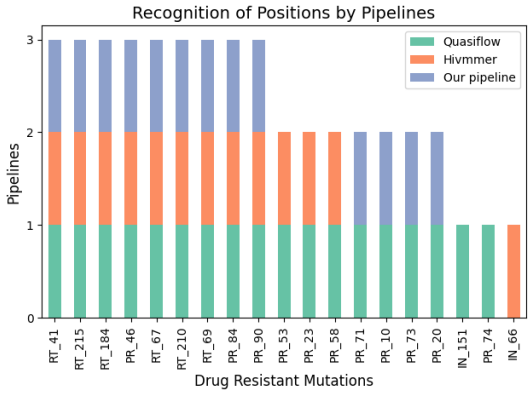

Figure 4.1: The changes in amino acids were recognized by three pipelines. Quasiflow was able to identify more changes, while our pipeline failed to detect significant mutations, such as those at positions 23 and 53 in the protease region.

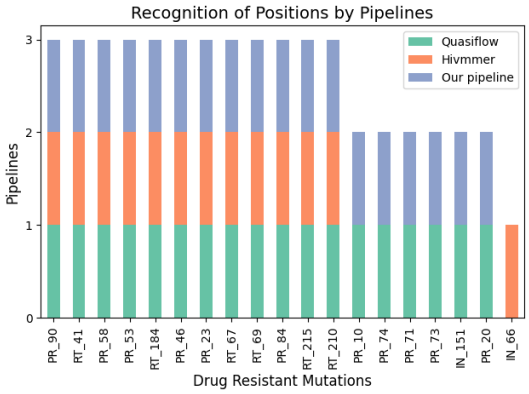

Figure 4.2: The changes in amino acids were recognized by three pipelines. After refining the HIV pipeline, our pipeline was able to identify the previously missed mutations. The results from our pipeline are now very similar to those from the Quasiflow pipeline.

As illustrated in Figure 4.2, after improving the HIV pipeline, the results from our pipeline are now very similar to those from the Quasiflow pipeline, and our pipeline was able to identify the mutations that it had failed to recognize in the first run. The pipeline can identify whether a mutation is present in only 0.9% of viral sequences. According to the equation 3.2 and the equation 3.1, setting minimum mapping quality and minimum base quality to 20 means that the variant calling considers only the bases and reads for which there is 99% confidence in their accuracy. By adjusting the appropriate cutoffs and ensuring that a mutation is identified in the patient's sample by at least two pipelines, we gain confidence in the presence of that mutation in the patient's sample.

To use Word2Vec for representing the samples with vectors of a certain dimension, it is necessary to determine the optimal dimension size as well as the optimal k-mer size. A balanced dataset consisting of four HIV-1 subtypes, A1, B, C, and D, with 400 observations was obtained from the Los Alamos National Laboratory HIV database [12]. The optimal dimensions of the vectors and the optimal k-mer size

were determined by plotting the accuracy obtained from the KNN classifier against the lengths of the k-mers and the dimensions of the vectors. The results are shown in Figure 4.3. The KNN classifier achieves the highest accuracy with k-mers of length five, six and seven. We decided to divide our k-mers into lengths of six and represent each k-mer with a vector of 250 dimensions.

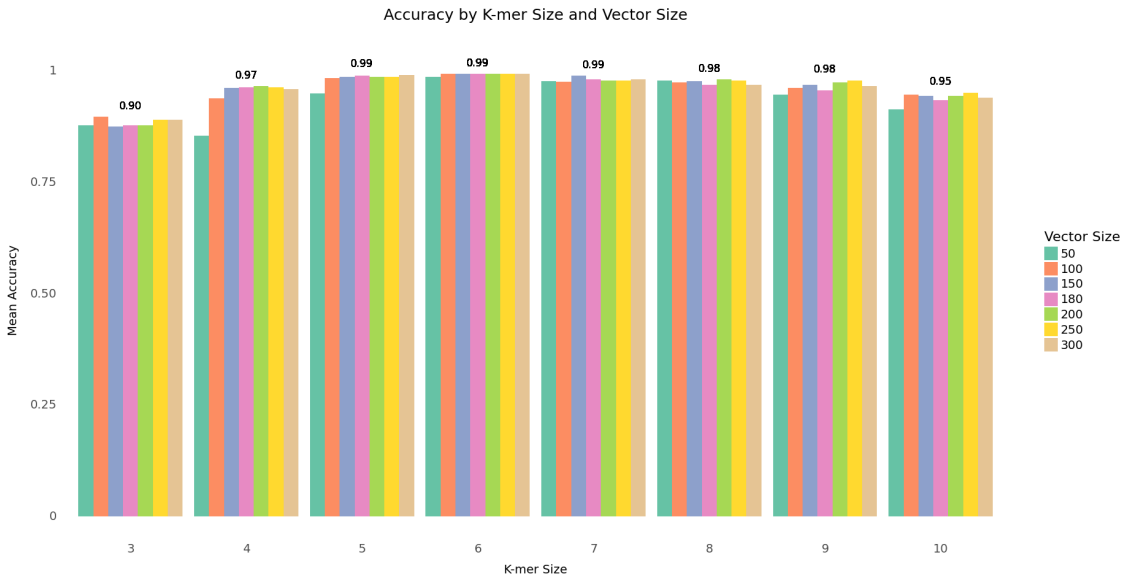

Figure 4.3: The optimal k-mers and vector size are obtained after calculating the mean accuracy of the KNN classifier. The 'K' value in KNN, representing the number of nearest neighbors, is set to five.

The data used for subtyping in this part is the same as that used in the previous project for subtyping. A balanced dataset consisting of four HIV-1 subtypes, A1, B, C, and D, with 400 observations was obtained from the Los Alamos National Laboratory HIV database [12]. The data was divided into k-mers of length six, and each k-mer was represented by a vector of dimension 250. After that, all vectors for all k-mers in each sample were averaged out, so that each sample was represented by a single vector of dimension 250. The different classifiers performance are presented in Figures 4.4, 4.5, 4.6, 4.7 and 4.8.

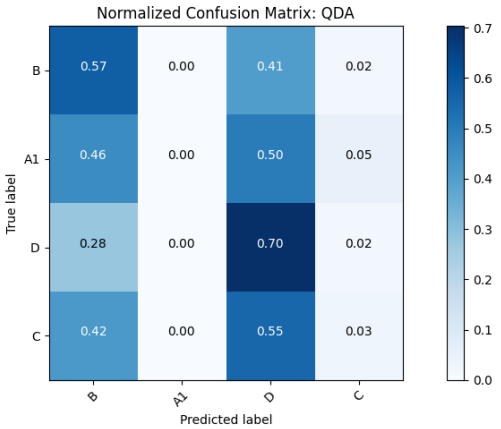

Figure 4.4: QDA performs the worst between the classifiers. It has mean accuracy of 32.68%, mean precision of 33.57%, mean recall score of 63.69% and mean F1-score of 43.16%

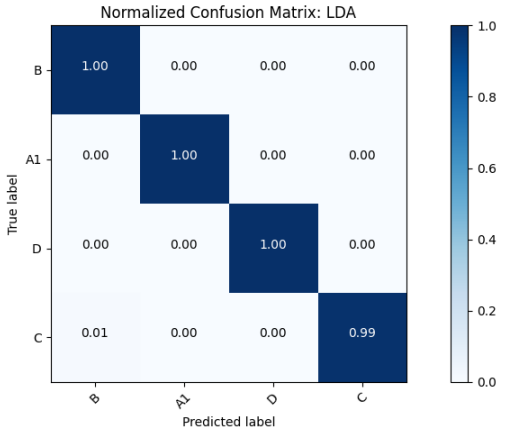

Figure 4.5: LDA performs really well. It has mean accuracy of 99.76%, mean precision of 99.77%, mean recall score of 99.76% and mean F1-score of 99.75%

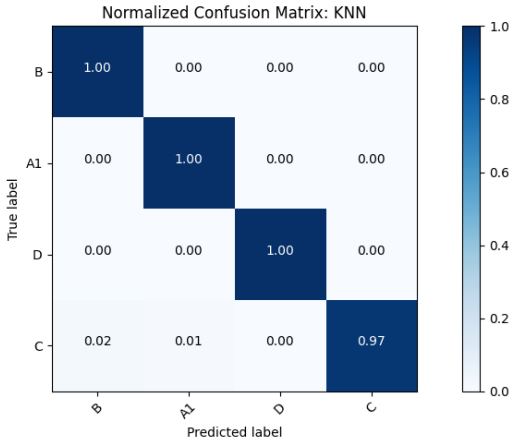

Figure 4.6: KNN performs really well. It has mean accuracy of 99.26%, mean precision of 99.32%, mean recall score of 99.26% and mean F1-score of 99.26%. The 'K' value, representing the number of nearest neighbors, is set to five.

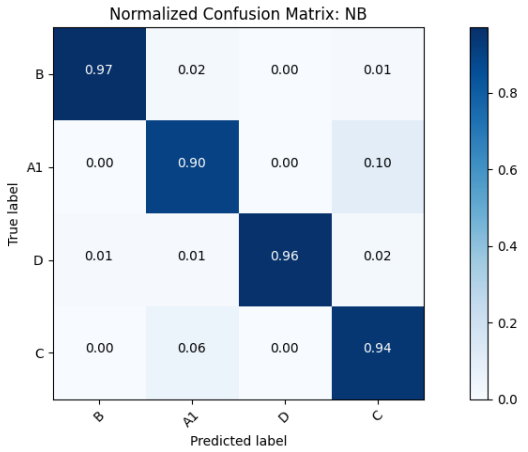

Figure 4.7: NB performs well. It has mean accuracy of 94.31%, mean precision of 94.79%, mean recall score of 94.31% and mean F1-score of 94.35%

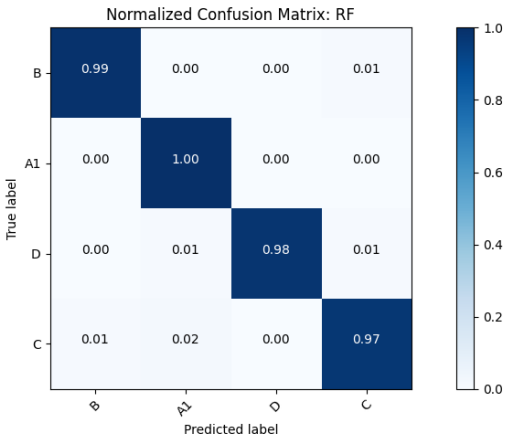

Figure 4.8: RF performs well. It has mean accuracy of 98.53%, mean precision of 98.71%, mean recall score of 98.53% and mean F1-score of 98.52%

Later, the same method is used on imbalanced data. Data consisting of 20 HIV-1 subtypes, with 1149 observations, is obtained from the Los Alamos National Laboratory HIV database [12]. The data is divided into k-mers of length 6, and each sample is represented by a vector of 250 dimensions. The subtypes G, 02_AG, 01_AE, A6, B, A1, C, and D have the most observations, with 101 observations each. The subtypes 103_01B and H have the fewest observations, with 10 observations each. The results from the classifiers RF and LDA are presented in Figures 4.9 and 4.10. To see the results of classifiers KNN, NB, and QDA, refer to Figures A.1, A.2 and A.3 in Appendix section.

Considering the classification results obtained from both balanced and imbalanced datasets, all the classifiers, except for QDA, were able to capture the characteristics of each patient's sample from these vectors of dimension 250. Comparing the results of the two methods applied for converting sequences into numerical values, we can see that TF-IDF was a better method for this purpose, as all the classifiers achieved higher metrics using this method. To see the results of the performance of different classifiers using TF-IDF, refer to Appendix Figures A.4, A.5, A.6, A.7, and A.8 for the balanced dataset, and Figures A.9, A.10, A.11, A.12, and A.13 for the imbalanced dataset.

By dividing the sequences into k-mers of a certain length and then assigning each k-mer a value based on its abundance and importance across different samples using TF-IDF, we define clear features for each sample. Classifiers, by mapping these features to each subtype, can learn a pattern from the training data and thus make more accurate predictions on the test data. The Word2Vec method is more suitable for very long sequences when the goal is to project them onto lower dimensions. However, this method is not suitable for viruses like HIV and HCV, as the longest sequence for these viruses is a maximum of 10000 nucleotide.

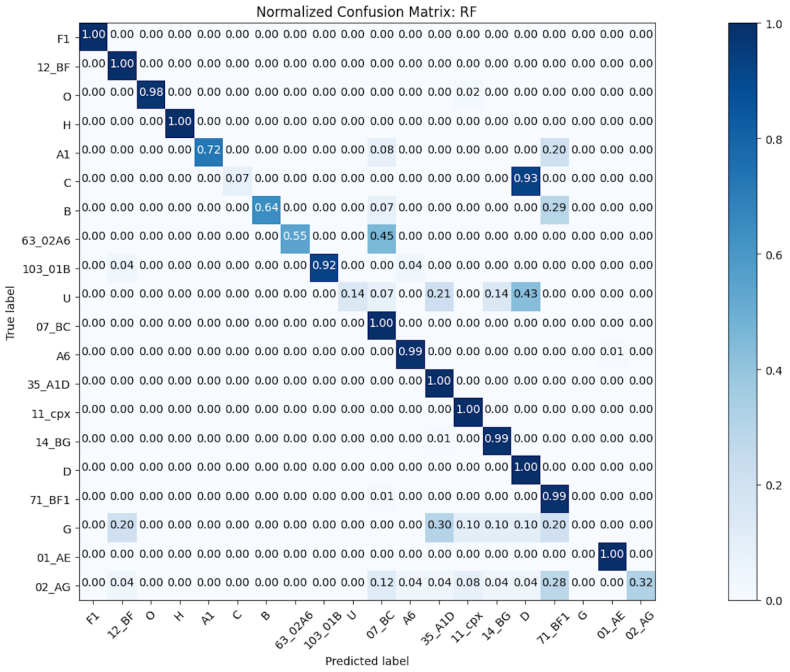

Figure 4.9: The performance of RF becomes slightly worse on imbalanced data. It has a mean accuracy of 93.04%, mean precision of 94.29%, mean recall score of 96.57% and mean F1-score of 94.50%.

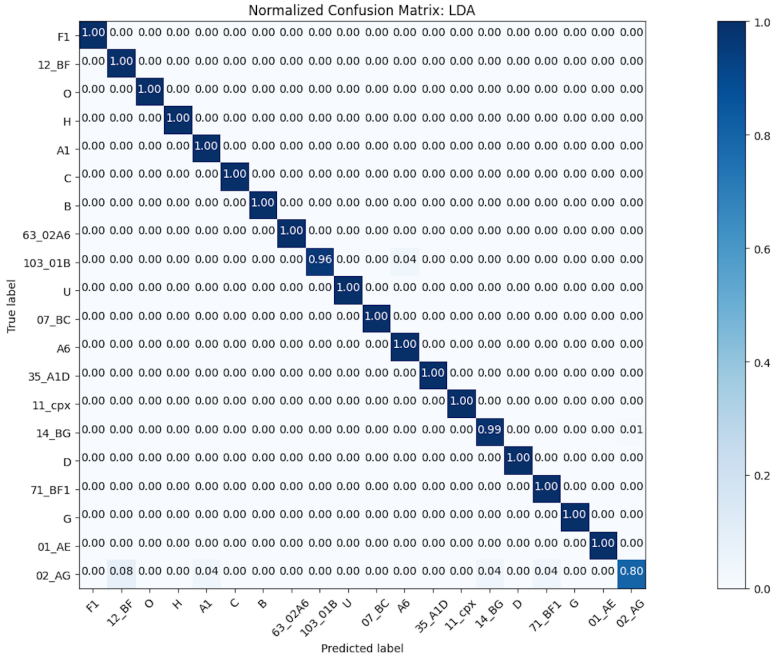

Figure 4.10: LDA performs the best among all the classifiers on imbalanced data. It has a mean accuracy of 99.39%, mean precision of 99.45%, mean recall score of 99.39% and mean F1-score of 99.36%.

4.2 Identifying low-frequency mutations in viral samples

The data for identifying low-frequency mutations belongs to a study conducted in France. The data was gathered from January 2014 to March 2017 at the Pitié-Salpêtrière, Saint-Antoine, and Bichat hospitals in Paris. 134 Illumina samples of patients that INSTI treatment failed for them were collected from NCBI with accession number SRP137063 [49]. The samples were run into the pipeline that we developed before for identifying low-frequency DRMs. The same samples were run into Quasiflow (nextflow version: v.23.10.0) pipeline as a validation step. The majority mutations found by our pipeline, Quasiflow and found in the study, "Prevalence and clinical impact of minority resistant variants in patients failing an integrase inhibitor-based regimen by ultra-deep sequencing", [1] is shown in Figure 4.11.

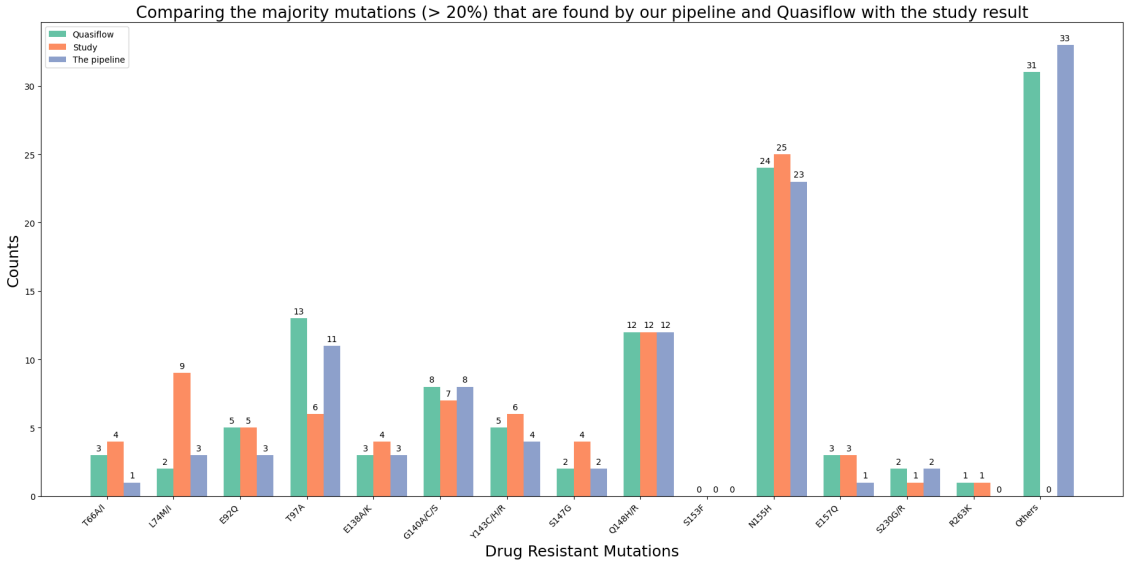

Figure 4.11: The majority DRMs found by our pipeline, Quasiflow and the study that the data belongs to [1].

The study analysed the samples with help of the company SmartGene, which is a bioinformatics company base in North America [52]. Previously, we defined mutations as minor if they are present in less than 20% of the viral sequences in a sample. A mutation is considered to be a majority variant if it constitutes more than 20% of the viral sequences in a sample. Analyzing the majority mutations found by all three methods presented in Figure 4.11, we observe that the results from our pipeline are more or less similar to those obtained from Quasiflow.

The Figure 4.12 shows the low-frequency DRMs that are found by our pipeline, Quasiflow and the study that the data belongs to.

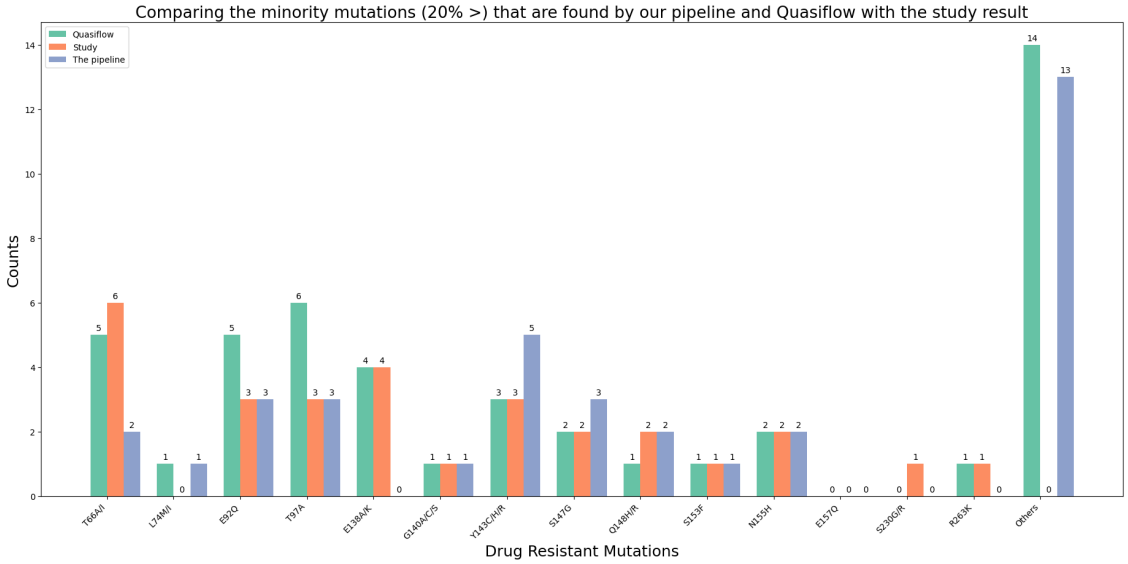

Figure 4.12: The minority mutations found by our pipeline, Quasiflow and the study that the data belongs to [1].

Analyzing the low-frequency mutations identified by the three pipelines in Figure 4.12, we notice more differences. Our pipeline found only two low-frequency DRM T66A\I, but the results obtained from SmartGene and Quasiflow are more similar. The significant difference between the results of our pipeline and the other two pipelines concerns the mutation E138A\K. Both Quasiflow and the study found four low-frequency mutations E138A\K, but our pipeline failed to find any. We decided to run the samples with lower base and mapping quality to see if we could identify these mutations. By lowering the base and mapping quality to 10, which implies that we have a 90% confidence in the correct alignment of the read and that the base nucleotide is correct, we were able to identify one low-frequency mutation E138A\K, one more T66A\I, and one R263K.

Since low-frequency mutations are present in only a very small fraction of viral samples, setting a proper threshold for coverage is crucial. Because these mutations are very rare, sometimes they cannot achieve high sequencing depth and may be missed if the cutoff for coverage is too high. By setting the minimum coverage to 100, we ensured that the reads with low sequencing depth are going to be analyzed in the pipeline.

The trade-off between analysing high quality data, while not missing existing mutations of lower quality, presents a significant challenge in analyzing NGS data. Differentiating between true and false positives is difficult, but setting a high threshold for the base and mapping quality, ensures that the captured mutations, whether majority or minority, are indeed true positives.

As illustrated in Figures 4.11 and 4.12, there are mutations found by both our pipeline and Quasiflow that are not reported in the study. These mutations and their descriptions are presented in Table A.1 in the Appendix section. According to Table A.1, most of these mutations have minimal effects on INSTI treatment, but

some, if they are combined with other mutations, can decrease the effectiveness of INSTI. For example, the low-frequency mutation L68V is presented together with the mutation E92Q in sample 21. If the mutation L68V is combined with E92Q, it can reduce the effectiveness of INSTI. The low-frequency mutation G149A was accompanied by the mutations G140S and Q148H in sample 56. G149A, along with mutations G140S and Q148H, can lead to treatment failure. The mutation V151I, which was not studied in the study, was reported as a majority mutation in six samples.

In this project, all 134 patients whose samples were analyzed failed INSTI treatment. One of the first assumptions when a treatment fails is that the viral sequences in the patient's body have mutations that make the virus resistant to the drug. Out of the 134 samples, no DRMs were found in 56 samples by either our pipeline or Quasiflow. Although no DRMs were detected, these 56 patients did not respond to INSTI treatment. This can be due to many reasons, such as the genetic makeup of each individual, which can cause them to respond differently to the treatment. However, one of the main reasons is that patients did not take the medication consistently and exactly as prescribed, leading to the virus not being fully suppressed.

4.3 Phylogenetic analysis

In this section, the results from building a phylogenetic tree using MEGA (v.11) and the results from studying evolutionary patterns using machine learning methods are presented.

4.3.1 Phylogenetic Tree

A dataset containing four viruses, HIV-1, HIV-2, SHIV, and SIV, along with their corresponding subtypes, was given to MEGA software (v.11) to construct a phylogenetic tree. Not all HIV-1 subtypes are included in the dataset to avoid making the tree more complicated. A phylogenetic tree was constructed using the MLE method and is presented in Figure 4.13. MEGA (v.11) by using MLE compares different evolutionary hypotheses, meaning, different evolutionary trees to see which one best explains the observed genetic data. This method has an iterative process to choose the best fit between many possible trees.

The tree starts with two main branches. The right branch contains all HIV-1 subtypes along with some SIV found in certain ape species, and the left branch comprises HIV-2 along with its corresponding subtypes, SHIV and some SIV in different species. As mentioned in the theory, the genetic similarity between HIV-1 and HIV-2 is only 55% and the tree clearly demonstrates that although HIV-1 and HIV-2 both lead to AIDS and have the same hosts, they are genetically very different.

The tree clearly supports the theory that HIV-1 is originated from either SIV in chimpanzees or SIV in gorillas. The genetic similarity of SIV in chimpanzees and gorillas to HIV-1 is greater than the genetic similarity between the two main types of HIV. In the left branch of the tree, it is also obvious that HIV-2 is genetically

very similar to SIV, which infects a monkey called Sooty mangabey. This result is also supported by the theory.

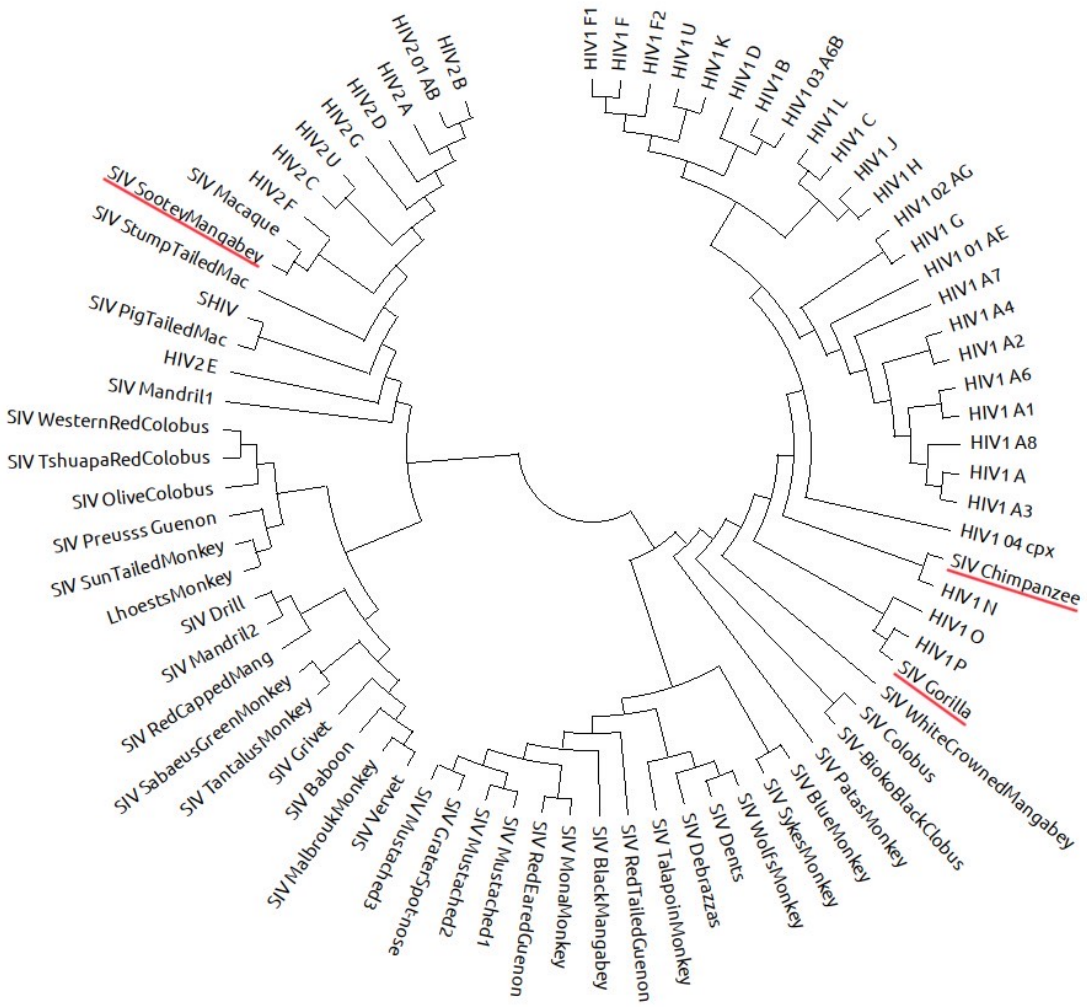

Figure 4.13: The phylogenetic tree illustrates the relationships between HIV-1, HIV-2, SHIV, and SIV. This tree was constructed in MEGA (v.11) using the MLE.

4.3.2 Predictive Evolutionary Analysis

Two feature selection methods, LASSO and Elastic Net, are used to identify the most significant codons for HIV subtyping. The data for this task was obtained from the Los Alamos National Laboratory and were used for the subtyping task [12]. The optimal α parameter for LASSO is 0.07 which was obtained after 10-fold cross validation with 1000 iterations and is presented in Figure 4.14. Obtaining a value of 0.07 for α indicates that there is some complexity in the data, and the data can be further simplified by setting some features, which may have been redundant, to zero. This means that, in our case, not all codons are informative for subtyping, and some of them can be simply omitted in our model. Having 0.07 as the optimal value for α also indicates that the model is neither too simple nor too complex, and

a balance has been found.

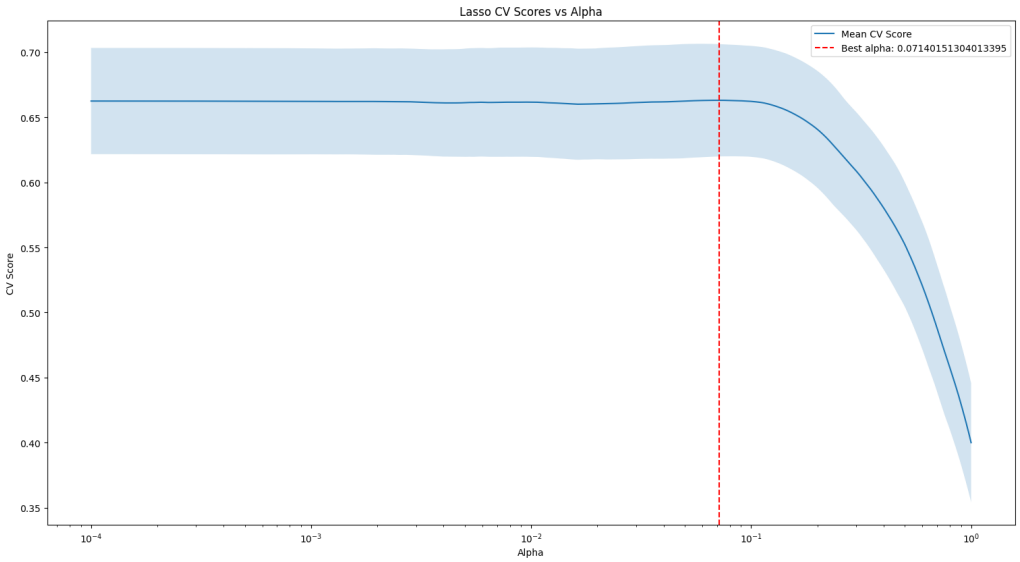

Figure 4.14: The optimal α parameter was obtained for LASSO through 10-fold cross validation and 1000 iterations.

After tuning the α parameter, the model was run with the best obtained parameter on the test dataset. The most important codons recognized by LASSO are presented in Figure 4.15. As illustrated in Figure 4.15, the coefficients of 25 codons are set to zero, indicating that these codons have no effect on identifying new subtypes.

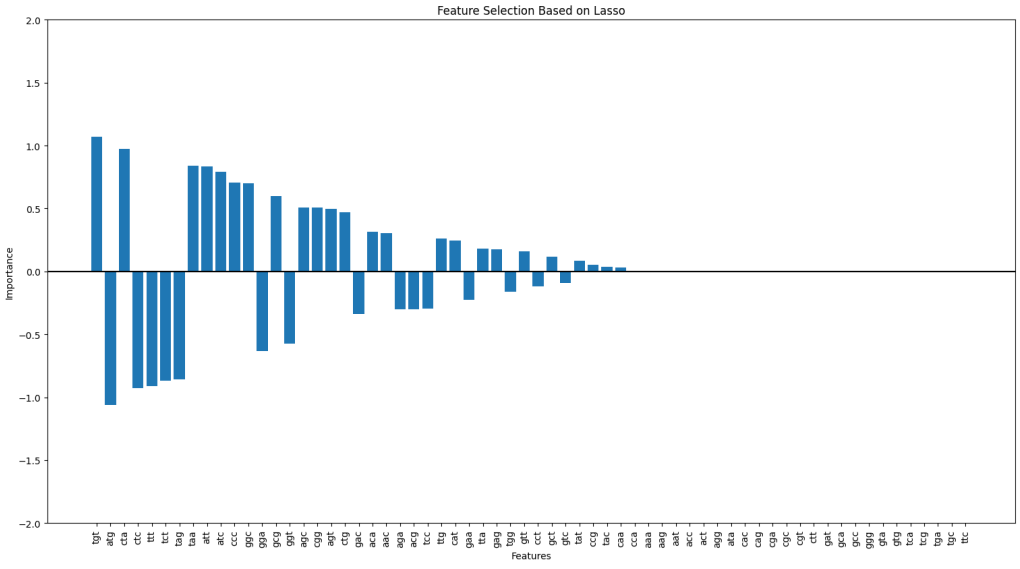

Figure 4.15: The most important codons found by LASSO regression.

The top five codons that have the highest negative coefficients in LASSO regression are: atg, ctc, ttt, tct and tag. The codons with positive coefficients are relevant in identifying new subtypes. The codons with negative coefficients have an inverse relationship with the outcome, meaning they are inversely related to the identification of new subtypes. When the codons with negative coefficients are present or their frequency is increased, the likelihood of identifying a new subtype decreases.

The most important codons for subtyping, with positive coefficients, are: tgt, cta, taa, att, and act. The codon tgt codes for Cysteine (Cys) amino acid, cta codes for Leucine (Leu), att codes for Isoleucine (Ile), act codes for Threonine (Thr) and taa is a stop codon. The change in amino acids Leu and Ile can be related to the mutations M41L and M184I. The mutation M41L which is the change from the amino acid Methionine to Leucine and the M184I, which is the change from the amino acid Methionine to Isoleucine, are more common in subtype B [53].

After studying the most relevant codons for subtyping using LASSO regression, we aimed to also use Elastic Net to study codons. There are two parameters for Elastic Net that need to be tuned. The optimal α as well as the optimal L1 ratio are determined for Elastic Net after 10-fold cross-validation with 500 iterations. The optimal α and the best L1 ratio are presented in Figure 4.16, with the values 0.09 and 0.20. Having these values means that the model tends to lean more toward Ridge regression, as indicated by the low L1 ratio. The model aims at shrinking some of the coefficients but not completely removing them by making them zero. These parameters help to build a model that is neither too complex nor too simple.

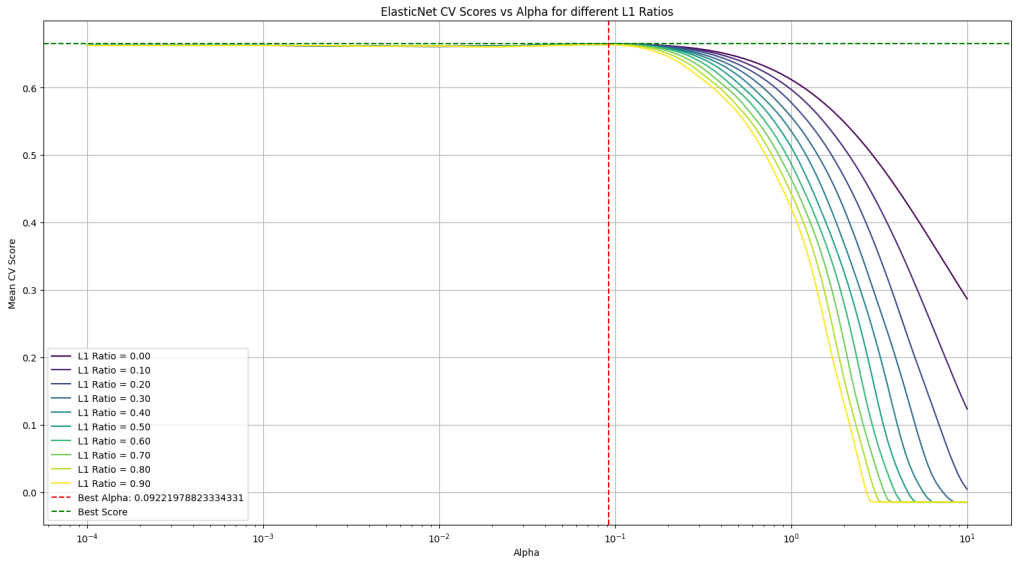

Figure 4.16: The optimal α and L1 ratio were obtained for Elastic Net through 10-fold cross validation and 500 iterations.

The most important codons recognised by Elastic Net are shown in Figure 4.17. The most important codons with positive coefficients for subtyping recognised by

Elastic Net are: taa, tgt, atc, ggc and cta. Out of these five most important codons selected by Elastic Net, the codons taa, tgt and cta are common codons chosen as the top relevant codons for subtyping by both Elastic Net and Lasso regression. The codon atc codes for amino acid Isoleucine (Ile) and the codon ggc codes for amino acid Glycine (Gly).

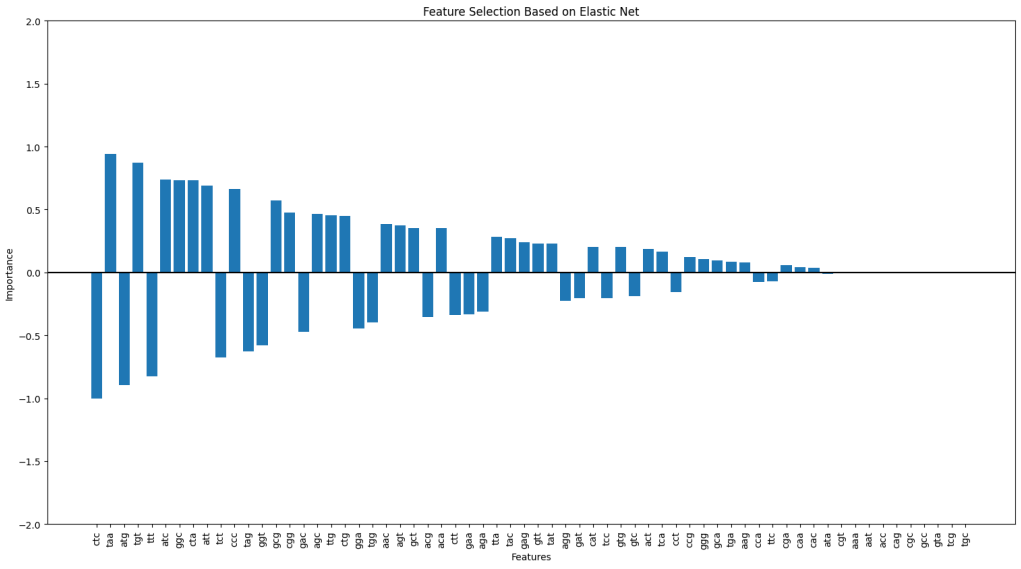

Figure 4.17: The most important codons found by Elastic Net regression.

After obtaining the most relevant codons for subtyping, all the sequences in the dataset were searched to see the specific positions of these codons in order to find a pattern. The aim was to see if these codons have a higher frequency in a specific part of the HIV-1 genomes across different subtypes, or if they are distributed evenly. No specific pattern was found to indicate that these codons are more common in a specific genomic region.

4.3.3 Identifying New Variants

Different classifiers have varying performances in identifying unseen and untrained data. The QDA classifier is completely confused and cannot accurately predict even the subtypes for which it was trained.

The RF classifier, in 10 predictions resulting from 10-fold cross-validation, could accurately predict the subtypes and CRFs A1, B, D, C, and 02_AG. Despite being trained to identify CRF 01_AE, the RF classifier mistakenly identified this CRF as subtype A1 seven out of ten times and correctly predicted it three out of ten times. Untrained CRF 06_cpx was predicted as subtype G nine out of 10 times and as subtype A1 once. Figure 4.18 shows the genetic composition of CRF 06_cpx. Considering the genetic composition of this CRF, it is evident that the majority sequence of this CRF belongs to subtype G. Although the RF classifier was not

specifically trained to identify 06_cpx, it could anyways capture the pattern and relate this CRF to the most similar subtype it was trained for.

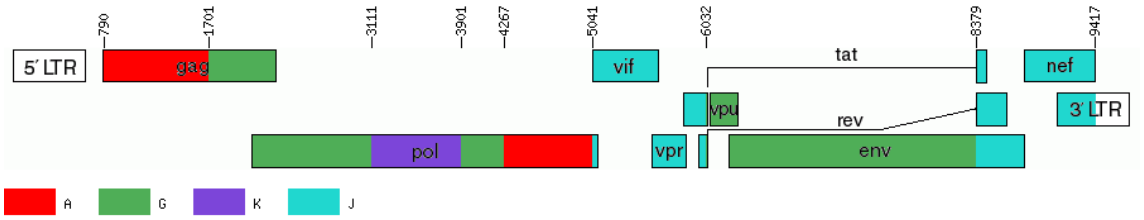

Figure 4.18: The genetic composition of CRF 06_cpx made by combination of subtype A, G, K and J. The Figure is from Los Alamos National Laboratory database [11].

The untrained CRF A1C was predicted as subtype C ten out of ten times by the RF classifier. Considering that this CRF is composed of a combination of subtypes A1 and C, the RF classifier again assigned this unknown CRF to the closest subtype that it was trained for. The untrained CRF 60_BC was predicted as subtype C ten out of ten times. According to Figure 4.19, the majority of this CRF belongs to subtype C, and again, the RF classifier was able to capture this pattern for the unseen data. RF obtained a mean accuracy of 69.17%.

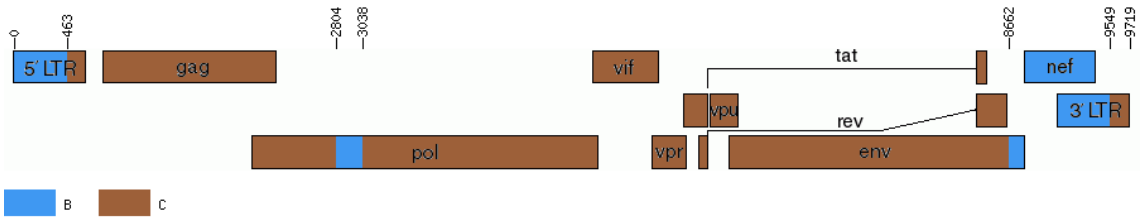

Figure 4.19: The genetic composition of a CRF 60_BC made by combination of subtype B and C. The Figure is from Los Alamos National Laboratory database [11].

The LDA classifier's performance when facing unknown data and patterns becomes significantly worse. Like the QDA, this classifier cannot even recognize the subtypes and CRFs it was trained for, achieving a mean accuracy of 31.67%. The KNN classifier's performance was even better than RF. It accurately identified all the subtypes and CRFs it was trained for. Like RF, it assigned untrained CRF 06_cpx to subtype G ten out of ten times. It also identified untrained CRFs A1C and 60_BC as subtype C ten out of ten times. Considering the genetic composition of CRFs 06_cpx in Figure 4.18 and CRF 60_BC in Figure 4.19, KNN assigned this unknown data to the most similar pattern it was trained for. KNN obtained a mean accuracy of 75%. The performance of the NB classifier is not particularly good.

This classifier occasionally fails to identify the subtypes and CRFs it was trained for, achieving a mean accuracy of 50.00%.

After testing the classifiers, the same data, in the form of a multi-fasta file, was given to MEGA (v.11) to construct a phylogenetic tree using the NJ method. A phylogenetic tree was constructed from 12 different subtypes and CRFs, which is illustrated in Figure 4.20. The values displayed at the top of each branch are bootstrap values. When MEGA (v.11) builds a phylogenetic tree using the NJ method, it resamples the original dataset with replacement to create random, smaller datasets. The bootstrap values varying between 0 to 100, indicate the percentage of times a particular branch appeared in different resampling. As illustrated in Figure 4.20, most of the branches have the highest confidence of 100, except for the branch containing A1C and CRF60_BC. This uncertainty arises because CRF A1C is composed of subtypes C and A1. The tree is not quite sure whether to cluster it with subtype C or with subtype A1.

Except for the cluster containing CRF60_BC, the NJ method in MEGA (v.11) by calculating the pairwise distances between different genomes was capable of identifying the subtypes and placing them in the correct branch. For example, the main left branch contains the subtype A1 and all CRFs that have A1 in their genetic composition.

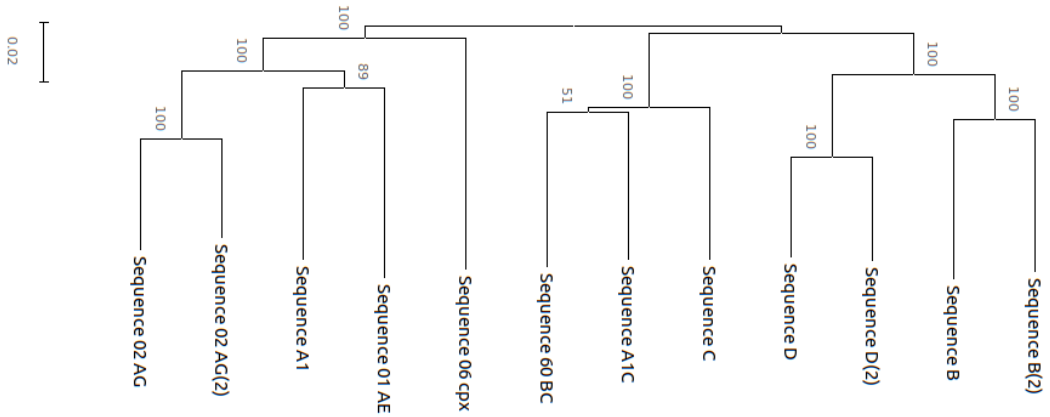

Figure 4.20: The phylogenetic tree is built using MEGA (v.11) with NJ method.

So far, we have tested two different methods for identifying subtypes. One method involved training various classifiers with labels to identify existing subtypes, while the other involved using methods like MLE and NJ in software like MEGA to build a phylogenetic tree. An idea for identifying new subtypes could be that if we receive a new and unknown sample, we first run it through the trained classifiers to see which subtype or CRF this new sample resembles. After labeling it with the outcome, we can run it in MEGA software alongside other subtypes, especially the subtype that the classifiers identified as the output of this new sample, as a validation step. In this way, we can determine if the classification was accurate and identify the correct

variant.

4.4 Subtyping HCV

A dataset containing four HCV subtypes was obtained from the Los Alamos National Laboratory HCV database [51]. The result of subtyping on test data is presented in Figures 4.21, 4.22, 4.23, 4.24 and 4.25.

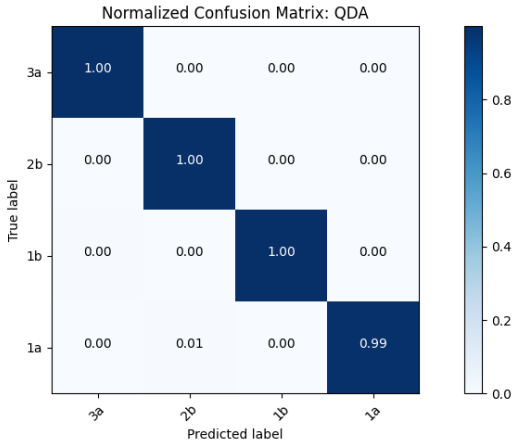

Figure 4.21: QDA performs really well. It has mean accuracy of 99.85%, mean precision of 99.85%, mean recall score of 99.85% and mean F1-score of 99.85%

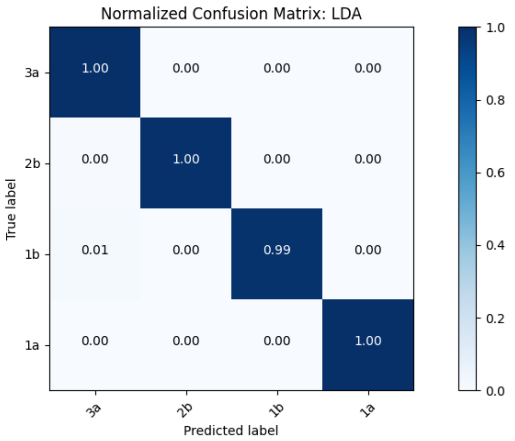

Figure 4.22: LDA performs really well. It has mean accuracy of 99.78%, mean precision of 99.78%, mean recall score of 99.78% and mean F1-score of 99.78%

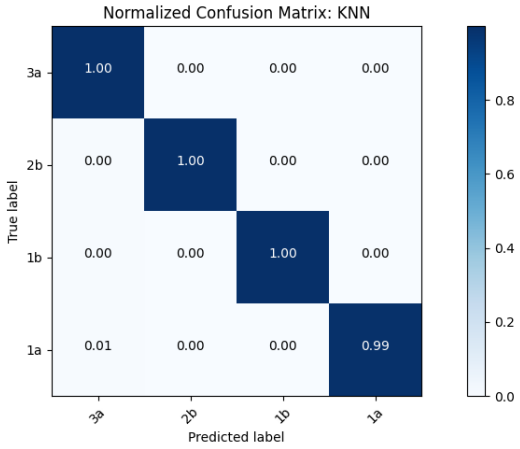

Figure 4.23: KNN performs really well. It has mean accuracy of 99.74%, mean precision of 99.74%, mean recall score of 99.74% and mean F1-score of 99.74%. The 'K' value, representing the number of nearest neighbors, is set to five.

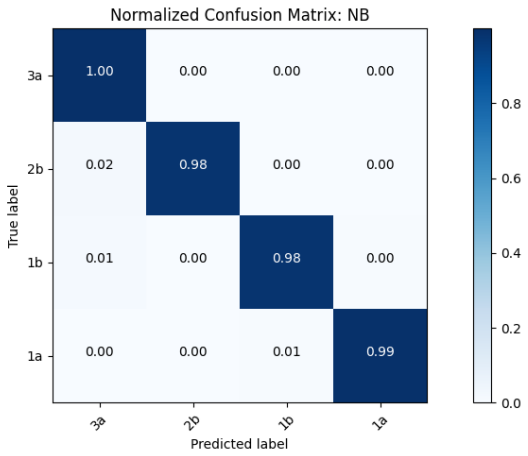

Figure 4.24: NB performs very well. It has mean accuracy of 99.37%, mean precision of 99.38%, mean recall score of 99.37% and mean F1-score of 99.37%

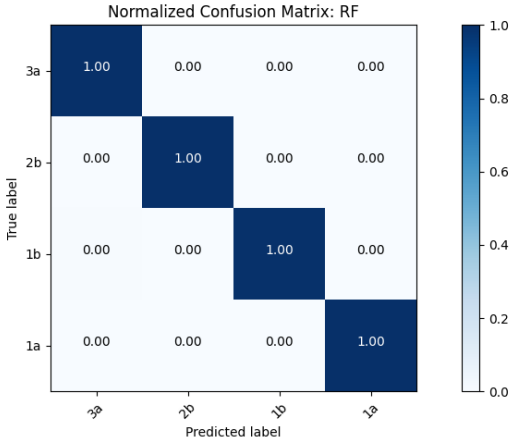

Figure 4.25: RF performs very well. It has mean accuracy of 99.85%, mean precision of 99.85%, mean recall score of 99.85% and mean F1-score of 99.85%

As it is illustrated in Figures 4.21, 4.22, 4.23, 4.24 and 4.25, all the classifiers achieve very good accuracy and all other statistical metrics for subtyping HCV. An interesting result is that the classifier QDA, which struggled in the classification of both balanced and imbalanced HIV-1 data, performs really well in classifying HCV. As shown in Figure A.4 and Figure A.9, the classifier QDA is quite confused when placing the unseen HIV-1 data into the correct subtypes. However, we see that the QDA classifier's performance becomes very good in classifying HCV subtypes. The reason behind it cannot be in the genetic composition of the different viruses. The classifiers just look for patterns and try to identify relevant features and map them to the correct labels. It does not matter to them if this data belongs to HIV or HCV. The reason of QDA classifier's good performance on HCV dataset, can be in the number of features, the number of subtypes, and the number of samples for each subtype.

The data used for subtyping HIV-1, as shown in Figure A.4, contains four HIV-1 subtypes, with 400 observations, 100 observations for each subtype. The number of features, after dividing the sequences into k-mers of length 8, is approximately 60000. When the number of features significantly exceeds the number of samples, the performance of QDA worsens. To improve the performance of this classifier, the dataset was expanded to include 20 subtypes with 1149 observations. Principal Component Analysis (PCA) was then applied to this data, resulting in projection onto the first 25 principal components and a significant reduction in the number of features. This reduction improved the QDA's performance on HIV-1 data, as shown in Figure A.9.

The data used for subtyping HCV includes four different subtypes, with 2708 observations and approximately 60000 features. The number of observations for QDA is now 2708 for four subtypes, instead of 400 for four subtypes, giving the classifier enough data to practice and learn the patterns and map the features to the labels.

In conclusion, by comparing the performance of QDA in Figures 4.21, A.4, and A.9, it is evident that as the number of samples increases or the number of features decreases, the performance of the QDA classifier improves.

In general, classification algorithms are not specialized pipelines that function well exclusively for HIV and HCV. These algorithms can handle various types of data, like biological or financial data. The important part of this method is to create a dataset with enough observations for each label with clear and relevant features.

5

Conclusion

In conclusion, we were able to develop methods to identify low-frequency DRMs, perform subtyping, conduct phylogenetic analysis, and apply these methods to identify new viral variants. The pipeline developed in the previous project was improved for identifying low-frequency DRMs. Improving all the steps of the developed pipeline are important, but the variant calling step requires proper cutoffs to identify true positives and minimize noise and false positives.

Machine learning algorithms can be powerful tools for studying genomic data. The first challenge is translating the data into informative numerical values. We tried to accomplish this using the methods TF-IDF and Word2Vec. TF-IDF resulted in a better transformation from nucleotide to numbers. Another challenge, and even a limitation of using supervised machine learning algorithms like classifiers for subtyping, is that these models can only identify and predict the data they have been trained on. The classifiers cannot reliably predict new variants, but our results showed that they could still identify some patterns in completely new CRFs and relate them to the nearest subtype that they were trained for.

Constructing a phylogenetic tree is a tool for observing the relationship between different species and for understanding how various species are related to each other. The software like MEGA, using various methods like MLE and NJ, can calculate the distance between different strains, and the likelihood of these strains ending up on the same branches. We tried to use feature selection techniques to find some evolutionary patterns, such as which parts of the virus are more prone to mutate. Although we could identify the most relevant codons for subtyping, the method had some limitations. The most important codons for subtyping are distributed evenly across the HIV-1 genome, and no pattern could be identified. Perhaps we could see a more obvious pattern if we investigated the most important gene for subtyping instead of studying the most relevant codons for subtyping. Also more complicated and sophisticated methods could be applied to perform this task, for example, using a neural network with many layers.

All the methods developed for identifying DRMs, subtyping, constructing phylogenetic trees, and using machine learning to study relevant codons are not specialized pipelines exclusively for the HIV virus and could be applied to other viruses, such as HCV. Only the last step of the DRM pipeline, which uses a specific software, Sierralocal, that determines if the variants are drug resistant, is specific to HIV. This step can be omitted for other viruses, and the variants can manually be extracted

from the VCF file after the variant calling step.

The methods developed for identifying DRMs and subtyping are offline tools. This is important because these pipelines and methods can be used in low-income countries where there is no stable internet connection. Using offline methods for analysing patients samples can also help in preventing leaking the patients information online. Patients infected with HIV and AIDS are still dealing with stigma in many places in the world, so it is important to protect patient information.

The primary goal of this project was to utilize NGS data due to its capability to detect low-frequency mutations. All the sequences used for various tasks are NGS clones, except the data utilized for subtyping, feature selection, and building an evolutionary tree for HIV-1, HIV-2, SHIV, and SIV. The data used for these tasks are mixed clones. This approach was chosen because supervised machine learning requires large sets of labeled data, and not all HIV-1 and HCV subtypes and CRFs were available in the form of labeled NGS clones. Another reason was the unavailability of all SIV subtypes as NGS sequences, due to the challenges of sequencing animals living in the wild. In conclusion, for identifying DRMs, especially low-frequency mutations, it is essential to use NGS sequences to avoid missing out on minority variants. However, for training supervised algorithms, sequencing methods other than NGS can also be utilized. The machine learning algorithms aim to identify a general pattern across different subtypes and do not focus solely on very low-frequency mutations.

Bibliography

[1] T. Nguyen, D. B. Fofana, M. P. Lê, *et al.*, "Prevalence and clinical impact of minority resistant variants in patients failing an integrase inhibitor-based regimen by ultra-deep sequencing", *Journal of Antimicrobial Chemotherapy*, vol. 73, no. 9, pp. 2485–2492, Jun. 2018, ISSN: 0305-7453. DOI: 10.1093/jac/dky198. eprint: https://academic.oup.com/jac/article-pdf/73/9/2485/25523249/dky198.pdf. [Online]. Available: https://doi.org/10.1093/jac/dky198.

[2] M. Libbrecht and W. Noble, "Machine learning applications in genetics and genomics", *Nature Reviews Genetics*, vol. 16, pp. 321–332, 2015. [Online]. Available: https://doi.org/10.1038/nrg3920.

[3] S. Starling, "The levee breaks—initial reports of aids", *Nature journal*, 2018. [Online]. Available: https://www.nature.com/articles/d42859-018-00002-y.

[4] B. H. H. Paul M. Sharp1, "Origins of hiv and the aids pandemic", *Cold Spring Harb Perspect Med*, 2011. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3234451/.

[5] German Advisory Committee Blood (Arbeitskreis Blut), Subgroup 'Assessment of Pathogens Transmissible by Blood', "Human Immunodeficiency Virus (HIV)", *Transfus Med Hemother*, vol. 43, no. 3, pp. 203–222, May 2016, Epub 2016 May 9. PMID: 27403093; PMCID: PMC4924471. DOI: 10.1159/000445852.

[6] LabCE, *Function of HIV Genes*, Accessed: 2024-04-09., 2013. [Online]. Available: https://www.labce.com/spg48969_function_of_hiv_genes.aspx.

[7] W.-S. H. Kazushi Motomura Jianbo Chen, "Genetic recombination between human immunodeficiency virus type 1 (hiv-1) and hiv-2, two distinct human lentiviruses", *ASM*, vol. 82, no. 4, 2008. [Online]. Available: https://journals.asm.org/doi/10.1128/jvi.01937-07.

[8] P. Akahome. "Hiv-1 subtypes". Accessed: 2023-10-15. (2021), [Online]. Available: https://www.aidsmap.com/about-hiv/hiv-1-subtypes.

[9] B. B. Reed AC Siemieniuk and M. J. Gill, "Increasing hiv subtype diversity and its clinical implications in a sentinel north american population", *Cold Spring Harb Perspect Med*, vol. 24, no. 2, 2013. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3720001/.

[10] D. M. Smith, D. D. Richman, and S. J. Little, "HIV Superinfection", *The Journal of Infectious Diseases*, vol. 192, no. 3, pp. 438–444, Aug. 2005, ISSN: 0022-1899. DOI: 10.1086/431682. eprint: https://academic.oup.com/jid/article-pdf/192/3/438/2485780/192-3-438.pdf. [Online]. Available: https://doi.org/10.1086/431682.

[11] "Circulating recombinant forms (crfs)". Accessed: 2024-02-22. (), [Online]. Available: https://www.hiv.lanl.gov/components/sequence/HIV/crfdb/crfs.comp.

[12] "Hiv sequence database". Accessed: 2024-02-20. (), [Online]. Available: https://www.hiv.lanl.gov/components/sequence/HIV/search/search.html.

[13] S. Chen and T. Morgan, "The natural history of hepatitis c virus (hcv) infection", *Int J Med Sci*, vol. 3, no. 2, pp. 47–52, 2006, Epub 2006 Apr 1. PMID: 16614742; PMCID: PMC1415841. DOI: 10.7150/ijms.3.47.

[14] D. Revie and S. Salahuddin, "Human cell types important for hepatitis c virus replication in vivo and in vitro: Old assertions and current evidence", *Virol J*, vol. 8, p. 346, Jul. 2011, PMID: 21745397; PMCID: PMC3142522. DOI: 10.1186/1743-422X-8-346.

[15] World Health Organization, *Hepatitis c*, https://www.who.int/news-room/fact-sheets/detail/hepatitis-c, Accessed: 2024-02-21, 2023.

[16] C. Hedskog, B. Parhy, S. Chang, *et al.*, "Identification of 19 Novel Hepatitis C Virus Subtypes—Further Expanding HCV Classification", *Open Forum Infectious Diseases*, vol. 6, no. 3, ofz076, Feb. 2019, ISSN: 2328-8957. DOI: 10.1093/ofid/ofz076. eprint: https://academic.oup.com/ofid/article-pdf/6/3/ofz076/33590500/ofz076.pdf. [Online]. Available: https://doi.org/10.1093/ofid/ofz076.

[17] A. Urbanowicz, R. Zagożdżon, and M. Ciszek, "Modulation of the immune system in chronic hepatitis c and during antiviral interferon-free therapy", *Arch Immunol Ther Exp (Warsz)*, vol. 67, no. 2, pp. 79–88, Nov. 15, 2019, PMID: 30443787; PMCID: PMC6420452. DOI: 10.1007/s00005-018-0532-8. doi: https://doi.org/10.1007/s00005-018-0532-8. [Online]. Available: https://doi.org/10.1007/s00005-018-0532-8.

[18] F. Karabiber. "Tf-idf — term frequency-inverse document frequency". Accessed: 2024-02-26. (), [Online]. Available: https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/.

[19] I. Logunova. "Word2vec: Why do we need word representations?" Accessed: 2024-02-26. (2023), [Online]. Available: https://serokell.io/blog/word2vec.

[20] D. Meyer. "How exactly does word2vec work?" Accessed: 2024-02-26. (2016), [Online]. Available: https://davidmeyer.github.io/ml/how_does_word2vec_work.pdf.

[21] A. E. Gorbalenya and C. Lauber, "Phylogeny of viruses", *Reference Module in Biomedical Sciences*, vol. 2017, B978-0-12-801238-3.95723–4, 2017, Epub 2017 Jun 26. PMCID: PMC7157450. DOI: 10.1016/B978-0-12-801238-3.95723-4.

[22] J. I. Mark P. Zwart1 Anne Kupczok, "Predicting virus evolution: From genome evolution to epidemiological trends", *Frontiers in Virology*, vol. 3, 2023. DOI: https://doi.org/10.3389/fviro.2023.1215709.

[23] K. Tamura, G. Stecher, and S. Kumar, "MEGA11: Molecular Evolutionary Genetics Analysis Version 11", *Molecular Biology and Evolution*, vol. 38, no. 7, pp. 3022–3027, Apr. 2021, ISSN: 1537-1719. DOI: 10.1093/molbev/msab120. eprint: https://academic.oup.com/mbe/article-pdf/38/7/3022/38827102/msab120.pdf. [Online]. Available: https://doi.org/10.1093/molbev/msab120.

[24] J. Brooks-Bartlett. "Probability concepts explained: Maximum likelihood estimation". Accessed: 2024-02-29. (2018), [Online]. Available: https://towardsdatascience.com/%20probability-concepts-explained-maximum-likelihood-estimation-c7b4342fdbb1.

[25] S. Sagitov, *Statistical inference*, Course Compendium, Statistical Inference, Chalmers University of Technology and the University of Gothenburg, 2023.

[26] N. Saitou and M. Nei, "The neighbor-joining method: A new method for reconstructing phylogenetic trees", *Molecular Biology and Evolution*, vol. 4, no. 4, pp. 406–425, 1987. DOI: 10.1093/oxfordjournals.molbev.a040454.

[27] K. Menon. "Feature selection in machine learning: All you need to know". Accessed: 2024-02-29. (2024), [Online]. Available: https://www.simplilearn.com/tutorials/machine-learning-tutorial/feature-selection-in-machine-learning.

[28] D. Kumar. "A complete understanding of lasso regression". Accessed: 2024-03-10. (), [Online]. Available: https://www.mygreatlearning.com/blog/understanding-of-lasso-regression/.

[29] L. R. S. Definitionr. "A complete understanding of lasso regression". Accessed: 2024-03-10. (), [Online]. Available: https://www.statisticshowto.com/lasso-regression/.

[30] S. Dhumne. "Elastic net regression detailed guide!" Accessed: 2024-03-11. (), [Online]. Available: https://medium.com/@shruti.dhumne/elastic-net-regression-detailed-guide-99dce30b8e6e.

[31] E. K. Jacob Murel. "What is ridge regression?" Accessed: 2024-03-11. (), [Online]. Available: https://www.ibm.com/topics/ridge-regression.

[32] "What is ridge regression?" Accessed: 2024-03-11. (), [Online]. Available: https://www.engati.com/glossary/ridge-regression.

[33] Laleh Varghaei, *Studying Genetic Diversity and Evolutionary Pattern in Human Immunodeficiency Virus*, https://bitbucket.org/1928diagnostics/thesis_laleh_2023_2024/src/master/, Accessed: 2024-04-09, 2023-2024.

[34] "S-andrews/fastqc". Accessed: 2023-09-10. (), [Online]. Available: https://github.com/s-andrews/FastQC.

[35] S. Chen, Y. Zhou, Y. Chen, and J. Gu, "fastp: an ultra-fast all-in-one FASTQ preprocessor", *Bioinformatics*, vol. 34, no. 17, pp. i884–i890, Sep. 2018. [Online]. Available: https://doi.org/10.1093/bioinformatics/bty560.

[36] M. Piper, R. Khetani, M. Mistry, and W. G. Jihe Liu. "Alignment using bowtie2". Accessed: 2023-09-10. (), [Online]. Available: https://hbctraining.github.io/Intro-to-ChIPseq-flipped/lessons/04_alignment_using_bowtie2.html.

[37] E. Garrison. "Freebayes /freebayes". Accessed: 2023-09-10. (), [Online]. Available: https://github.com/freebayes/freebayes.

[38] H. Li, B. HandsakerPetr, and P. Danecek. "Bcftools(1) manual page". Accessed: 2023-09-10. (), [Online]. Available: https://samtools.github.io/bcftools/bcftools.html.

[39] J. C. Ho, G. T. Ng, M. Renaud, and A. F. Poon, "Sierra-local: A lightweight standalone application for secure hiv-1 drug resistance prediction", *bioRxiv*, [Online]. Available: https://www.biorxiv.org/content/10.1101/393207v2.

[40] European Bioinformatics Institute. "Ena browser - drr030218". Accessed: 2023-11-15. (2023), [Online]. Available: https://www.ebi.ac.uk/ena/browser/view/DRR030218.

[41] M. Howison. "Kantorlab /hivmmer". Accessed: 2023-09-10. (2020), [Online]. Available: https://github.com/kantorlab/hivmmer.

[42] A. Ssekagiri, D. Jjingo, I. Lujumba, *et al.*, "QuasiFlow: a Nextflow pipeline for analysis of NGS-based HIV-1 drug resistance data", *Bioinformatics Advances*, vol. 2, no. 1, Nov. 2022, ISSN: 2635-0041. [Online]. Available: https://doi.org/10.1093/bioadv/vbac089.

[43] H. Li, *Aligning sequence reads, clone sequences and assembly contigs with bwa-mem*, If you use the BWA-MEM algorithm or the fastmap command, or want to cite the whole BWA package, 2013. arXiv: 1303.3997v2 [q-bio.GN].

[44] A. Wilm *et al.*, "Lofreq: A sequence-quality aware, ultra-sensitive variant caller for uncovering cell-population heterogeneity from high-throughput sequencing datasets", *Nucleic Acids Research*, vol. 40, no. 22, pp. 11 189–11 201, 2012.

[45] Z. J. Johnson, D. D. Krutkin, P. Bohutskyi, and M. G. Kalyuzhnaya, "Chapter eight - metals and methylotrophy: Via global gene expression studies", in *Rare-Earth Element Biochemistry: Methanol Dehydrogenases and Lanthanide Biology*, ser. Methods in Enzymology, J. A. Cotruvo, Ed., vol. 650, Academic Press, 2021, pp. 185–213. DOI: https://doi.org/10.1016/bs.mie.2021.01.046. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0076687921000689.

[46] K. Goswami and N. Sanan-Mishra, "Chapter 7 - rna-seq for revealing the function of the transcriptome", in *Bioinformatics*, D. B. Singh and R. K. Pathak, Eds., Academic Press, 2022, pp. 105–129, ISBN: 978-0-323-89775-4. DOI: https://doi.org/10.1016/B978-0-323-89775-4.00002-X. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B978032389775400002X.

[47] SAMtools, *HTS-specs: SAM Format Specification and related HTS Format specifications*, Accessed: 2024-03-23, 2024. [Online]. Available: https://github.com/samtools/hts-specs.

[48] H. Gourlé, O. Karlsson-Lindsjö, J. Hayer, and E. Bongcam-Rudloff, "Simulating illumina data with insilicoseq", *Bioinformatics*, 2018. DOI: 10.1093/bioinformatics/bty630.

[49] National Library of Medicine. "Srp137063". Accessed: 2024-02-27. (2018), [Online]. Available: https://www.ncbi.nlm.nih.gov/sra/?term=SRP137063.

[50] N. C. Institute. "Codon". Accessed: 2024-03-19. (), [Online]. Available: https://www.cancer.gov/publications/dictionaries/genetics-dictionary/def/codon.

[51] "Hcv sequence database". Accessed: 2024-02-27. (), [Online]. Available: https://hcv.lanl.gov/components/sequence/HCV/search/searchi.html.

[52] SmartGene Services, Inc., *SmartGene - Simply Managing Complex Data*, https://www.smartgene.com/, Accessed: 2024-03-23, 2024.

[53] B. Nastri, P. Pagliano, C. Zannella, *et al.*, "Hiv and drug-resistant subtypes", *Microorganisms*, vol. 11, no. 1, p. 221, Jan. 2023. DOI: 10.3390/microorganisms11010221.

[54] D. Goodman, R. Hluhanich, J. Waters, *et al.*, "Integrase inhibitor resistance (elvitegravir/raltegravir) involves complex interactions among primary and secondary resistance mutations: A novel mutation l68v/i associates with e92q and increases resistance", in *Proceedings of the XVII International HIV Drug Resistance Workshop*, Reported by Jules Levin, XVII HIV Drug Resistance Workshop, Sitges, Spain: Gilead Sciences, Inc., Durham, NC, USA; Gilead Sciences, Inc., Foster City, CA, USA; Monogram Biosciences, Inc., South San Francisco, CA, USA, Jun. 2008.

[55] "Insti resistance notes". Accessed: 2024-03-19. (), [Online]. Available: https://hivdb.stanford.edu/dr-summary/resistance-notes/INSTI/.

[56] O. Goethals, R. Clayton, M. Van Ginderen, *et al.*, "Resistance mutations in human immunodeficiency virus type 1 integrase selected with elvitegravir confer reduced susceptibility to a wide range of integrase inhibitors", *J Virol*, vol. 82, no. 21, pp. 10 366–10 374, Nov. 2008. DOI: 10.1128/JVI.00470-08.

[57] "Insti resistance comments". Accessed: 2024-03-19. (), [Online]. Available: https://hivdb.stanford.edu/dr-summary/comments/INSTI/.

A

Appendix 1

In this section, the supplementary material and results are provided.

Mutation	Description
L68V	This mutation is not under surveillance, however, L68V can enhance drug resistance when combined with the E92Q mutation [54].
Q95K	Q95K has minor effect on INSTI treatment [55].
H114Y	This mutation is not under surveillance and does not cause treatment failure when comes alone [56].
A128T	A128T is a rare mutation that does not impact INSTI treatment [57].
E138D	E138D can appear in 1% to 2% of viral samples from patients for whom INSTI treatment has failed. It does not reduce the effectiveness of INSTI treatment [57].
G149A	G149A is a rare mutation that has no effect on INSTI treatment failure when it comes alone but can lead to treatment failure when combined with mutations at positions 140 and 148 [57].
V151I	V151I is a very rare mutation that significantly decreases the effectiveness of INSTI treatment [57].
G163R	This is an accessory mutation that can co-occur with the N155H mutation. It does not lead to INSTI failure [55].
S230N	This mutation does not lead to INSTI failure and is not considered a major concern [57].
D232N	D232N is a rare mutation with only a minimal effect on INSTI treatment [57].

Table A.1: The mutations that were not reported in the study but were identified by our pipeline and Quasiflow.

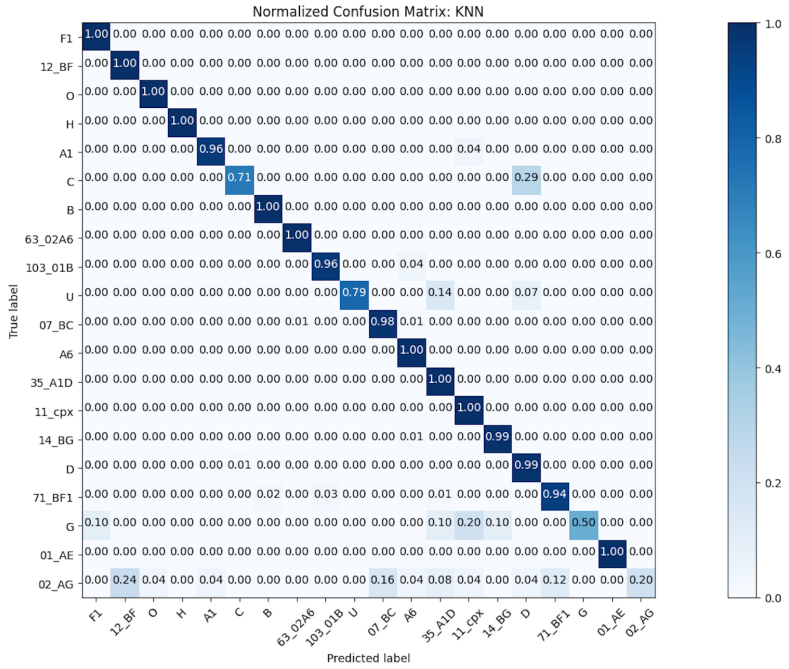

Figure A.1: The performance of KNN is good. It has a mean accuracy of 96.17%, mean precision of 96.99%, mean recall score of 97.37%, and mean F1-score of 96.68%. The 'K' value, representing the number of nearest neighbors, is set to five.

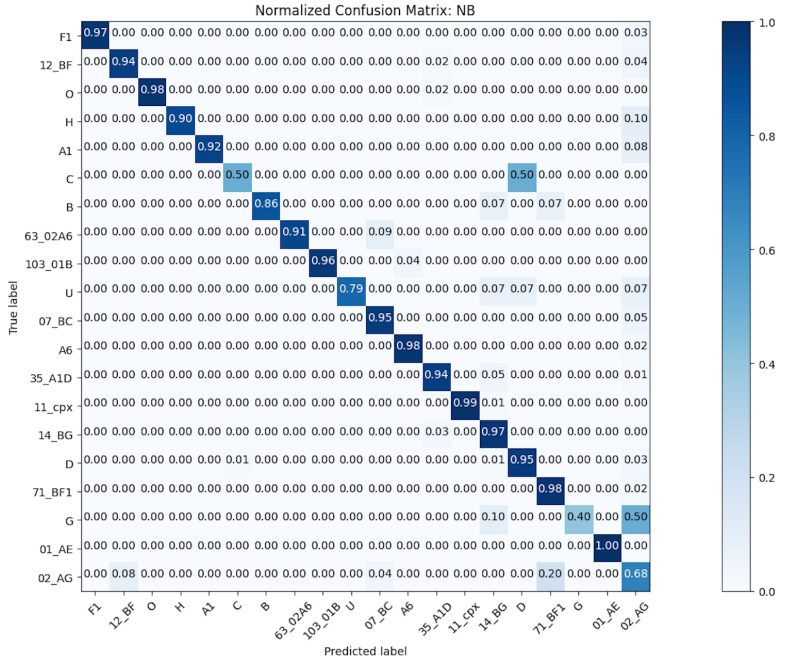

Figure A.2: The performance of NB is good. It has a mean accuracy of 94.34%, mean precision of 95.82%, mean recall score of 95.25%, and mean F1-score of 95.17%.

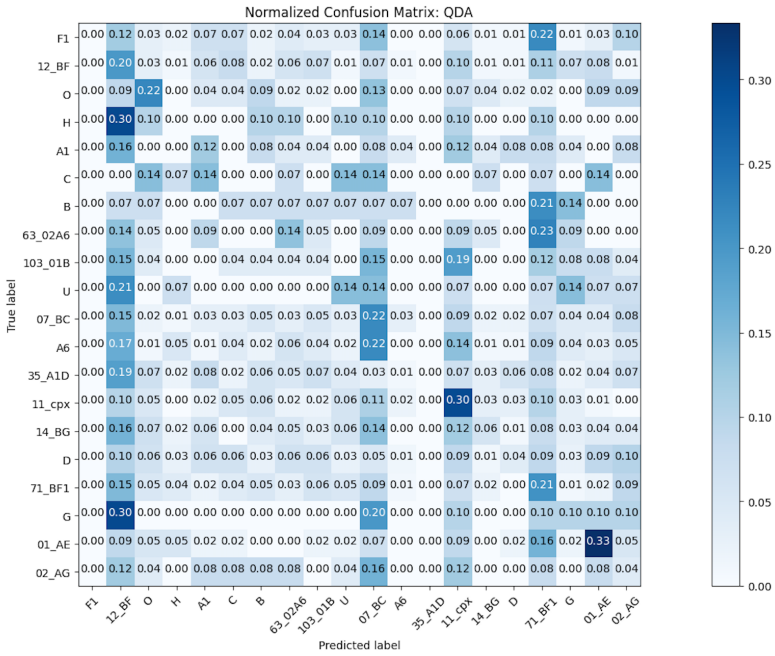

Figure A.3: The performance of QDA is very bad. The classifier is completely confused in identifying the correct subtypes. It has a mean accuracy of 12.45%, mean precision of 17.37%, mean recall score of 20.38%, and mean F1-score of 17.17%.

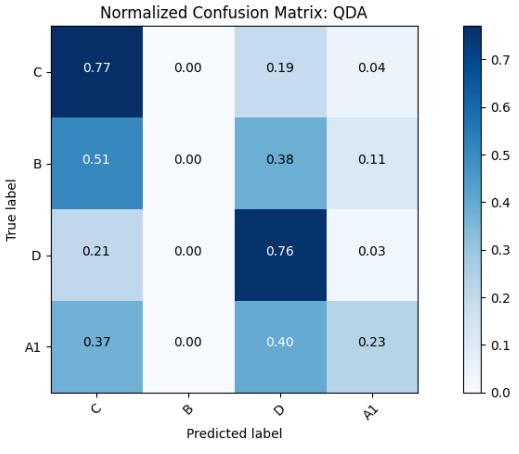

Figure A.4: QDA performs the worst between the classifiers. It has mean accuracy of 44.00%, mean precision of 54.24%, mean recall score of 77.90%, and mean F1-score of 58.16%.

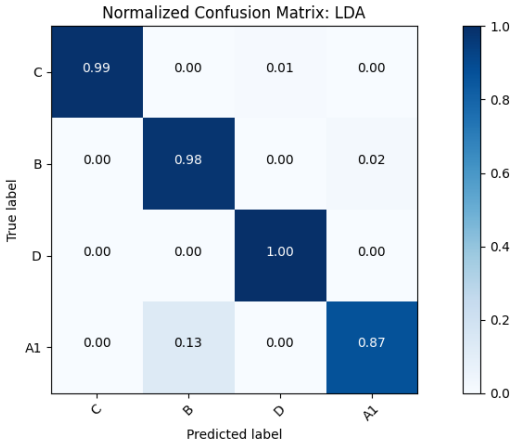

Figure A.5: LDA performs well. It has mean accuracy of 96.00%, mean precision of 96.63%, mean recall score of 96.00%, and mean F1-score of 96.02%.

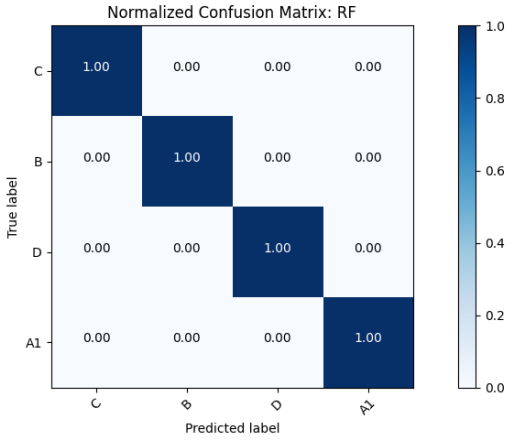

Figure A.6: RF performs really well. It has mean accuracy of 100.00%, mean precision of 100.00%, mean recall score of 100.00%, and mean F1-score of 100.00%.

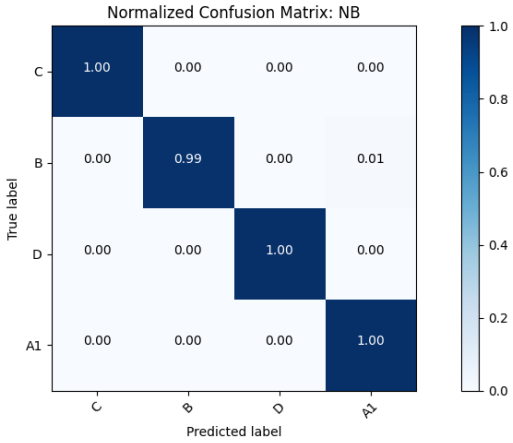

Figure A.7: NB performs really well. It has mean accuracy of 99.75%, mean precision of 99.78%, mean recall score of 99.75%, and mean F1-score of 99.75%.

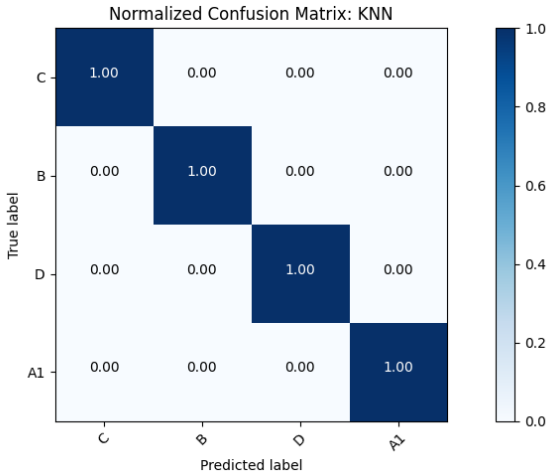

Figure A.8: KNN performs really well. It has mean accuracy of 100.00%, mean precision of 100.00%, mean recall score of 100.00%, and mean F1-score of 100.00%. The 'K' value, representing the number of nearest neighbors, is set to five.

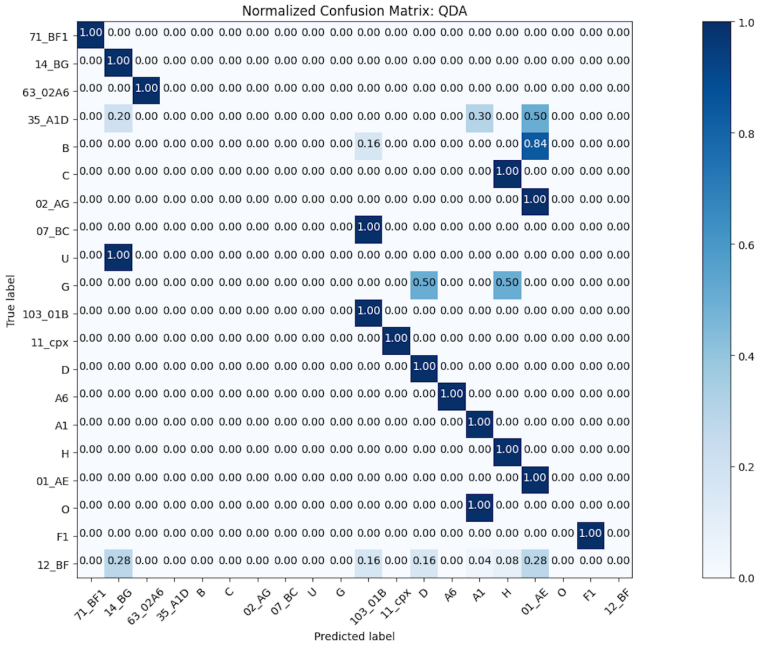

Figure A.9: QDA performance becomes better after PCA projection. It has mean accuracy of 86.08%, mean precision of 88.71%, mean recall score of 100.00%, and mean F1-score of 93.40%.

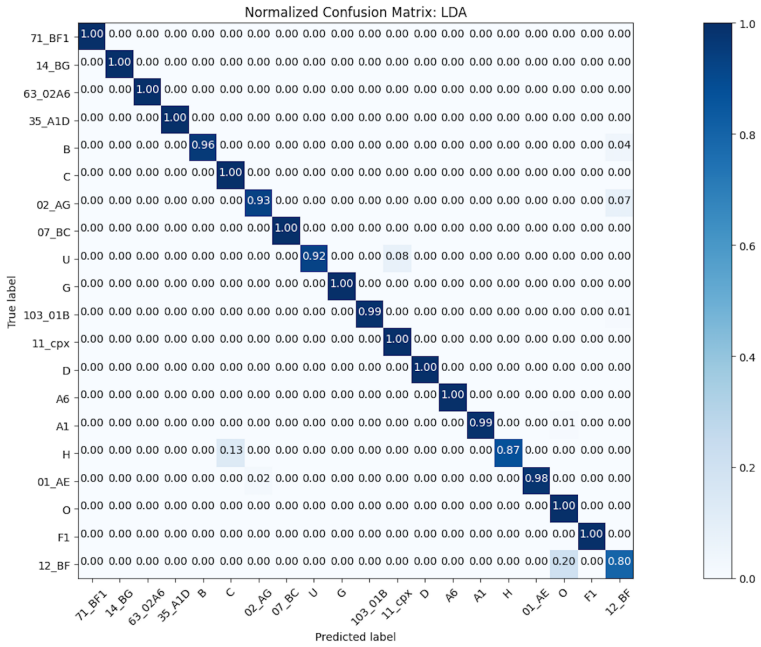

Figure A.10: LDA performs really well even after PCA projection. It has mean accuracy of 98.00%, mean precision of 99.09%, mean recall score of 98.08%, and mean F1-score of 98.41%.

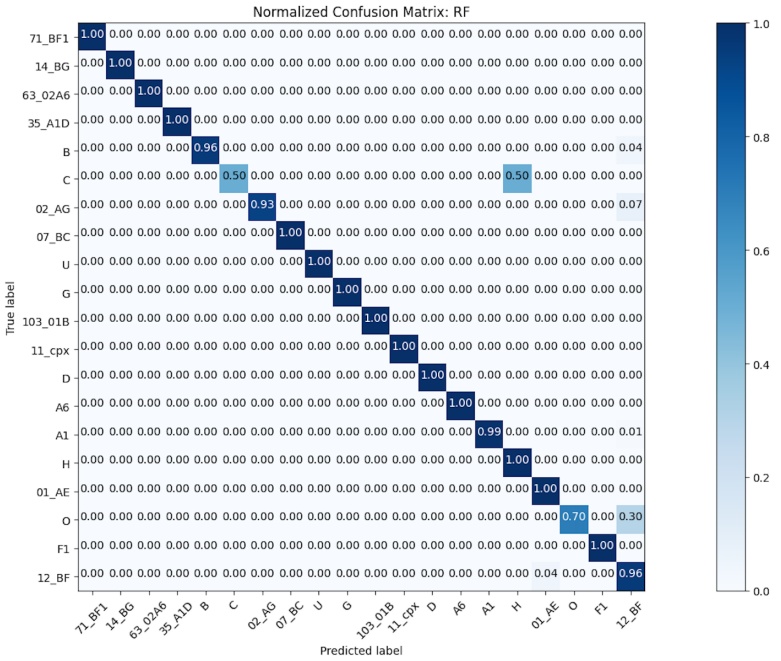

Figure A.11: RF has very good performance. It has some difficulty only with grouping subtype C and O. RF has mean accuracy of 98.78%, mean precision of 99.08%, mean recall score of 99.39%, and mean F1-score of 99.09%.

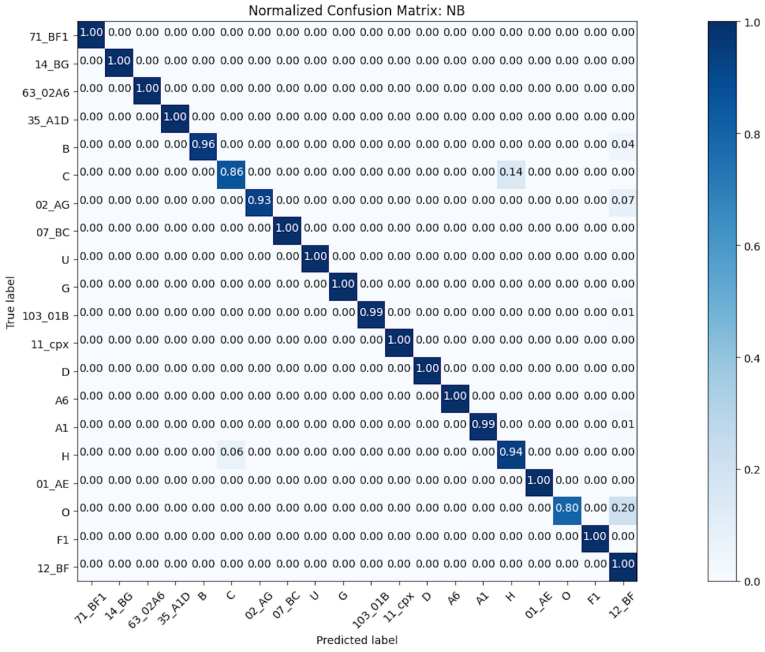

Figure A.12: NB performs really well. It has mean accuracy of 98.87%, mean precision of 99.42%, mean recall score of 99.04%, and mean F1-score of 99.10%.

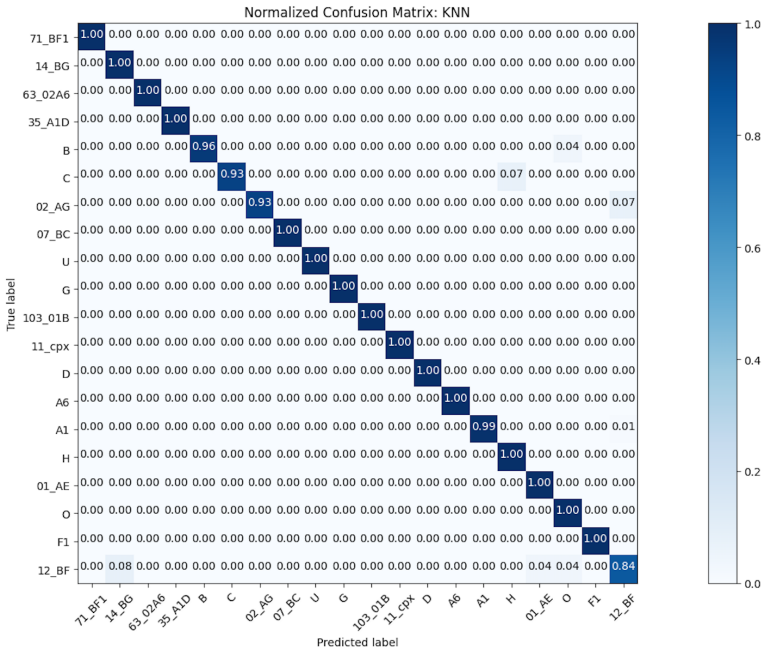

Figure A.13: KNN performs really well. It has mean accuracy of 99.30%, mean precision of 99.43%, mean recall score of 99.48%, and mean F1-score of 99.39%. The 'K' value, representing the number of nearest neighbors, is set to five.

DEPARTMENT OF MATHEMATICAL SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY