Universidad Autónoma de San Luis Potosí
Facultad de Ingeniería
Centro de Investigación y Estudios de Posgrado

Machine learning approaches for pattern recognition on genetic data from the Human Immunodeficiency Virus

Para obtener el grado de:
Doctorado en Ciencias de la Computación

Presenta:
José Salomón Altamirano Flores

Asesor:
Dr. Juan Carlos Cuevas Tello
Co-Asesor:
Dr. Christian Alberto García Sepúlveda

San Luis Potosí, S. L. P. Junio de 2020

This thesis is dedicated to my mother, my grand-mother, sisters and brothers.

Acknowledgements

This thesis would not have been possible without all the help and support of the kind people with whom I had the privilege to interact. Unfortunately, only some of them are particularly mentioned here.

Above all, I would like to thank my mother Elia for all her support and for being a great example for our family. I thank my sisters and brothers for their undoubted support throughout this process.

This thesis would not have been possible without the help, support and patience of my principal advisor Dr. Juan Carlos Cuevas Tello and my co-advisor Dr. Christian Alberto García Sepulveda. I thank Dr. Cuevas Tello for sharing his knowledge on Artificial Intelligence, data analysis and for his guidance on the development of this work. In a similar way, I thank Dr. García Sepúlveda for sharing his knowledge on the biomedical field and for all his helpful insights that made this work possible.

I also would like to thank Dr. José Ignacio Nuñez Varela and Dr. Cesar Augusto Puente Montejano support and important advice that helped in the improvement of this work.

I want to thank the Laboratorio de Genómica Viral y Humana for allowing me the opportunity to collaborate with them and for all the received support in the development of this work. In the same venue, I would like to thank the Universidad Autónoma de San Luis Potosí for this magnificent and challenging opportunity.

Finally, I want to thank the Consejo Nacional de Ciencia y Tecnología for the support that allowed this adventure.

Abstract

This thesis presents a new methodology for the assessment and pattern suggestion of clinical associations in HIV. The search for associations of genetic markers and clinical endpoints commonly is through a hypothesis-driven approach. However, exploring a high number of variable combinations as hypotheses involves statistical corrections for multiple testing, a strategy that decreases the chance of finding significant associations. In addition, the use of small datasets further hinders discoveries. Since machine learning approaches can work on non-linear problems with no pre-defined models, researchers have been using them in the search for associations. Our methodology combines five machine learning approaches. The Apriori, C4.5 and Multifactor Dimensionality Reductor provide distinct ways for discriminating between cases. Our methodology assesses the relevance of the genetic variables and improves their discovery using Artificial Neural Networks. Finally, ID3 can further substantiate the suggested associations for using in traditional statistical tests. Our methodology also can suggest the most important genetic variables associated with a clinical endpoint of the dataset, thus reducing the number of tested hypotheses. By applying this methodology on an HIV dataset of genetic viral traits coupled to clinical information for patients diminished the number of hypotheses from 172,186,880 to only 23. We found seven of these as significant associations after testing with the Fisher's exact test. Four associations suggest protection for the patient while the remaining three suggested risks to HIV progression. Although their biological impact requires further research, our methodology was useful for guiding the search for relevant novel hypotheses and at describing novel trait combinations determining clinical progression.

Resumen

Esta tesis presenta una nueva metodología para la evaluación y sugerencia de asociaciones clínicas en el VIH. La búsqueda de asociaciones entre marcadores genéticos y estados clínicos está comúnmente basada en un enfoque guiado por hipótesis. Sin embargo, explorar un gran número de combinaciones de variables como hipótesis requiere la aplicación de correcciones estadísticas derivado del uso de múltiples pruebas, lo cual disminuye las posibilidades de encontrar asociaciones estadísticamente significativas. Lo que es más, el uso de bases de datos pequeñas, con relativamente pocos ejemplos, dificulta aún más la posibilidad de hacer descubrimientos. Actualmente los enfoques de aprendizaje automático están siendo utilizados en la búsqueda de asociaciones dado que los enfoques funcionan en problemas no lineales, sin la necesidad de predefinir modelos. Nuestra metodología usa una combinación de cinco enfoques de aprendizaje automático. Los enfoques Apriori, C4.5 y el Reducción de la Dimensionalidad Multifactor (*Multifactor Dimensionality Reduction*) proveen distintas maneras de discriminar entre casos. Nuestra metodología mide la relevancia de las variables genéticas, lo cual es subsecuentemente mejorado usando el enfoque de Redes Neuronales Artificiales. Como paso final, el uso del algoritmo ID3 permite substanciar las asociaciones sugeridas como hipótesis para las pruebas estadísticas finales. Nuestra metodología es capaz de sugerir las variables genéticas más importantes asociadas con estados clínicos descritos en las bases de datos mientras reduce el número de hipótesis a probar. La aplicación de nuestra metodología a una base de datos relacionada con información genética descriptiva de características del VIH y estados clínicos de pacientes, permitió la reducción en el número de hipótesis a probar de 172,186,880 a sólo 23. Siete de estas asociaciones fueron estadísticamente significativas después de aplicar la prueba exacta de Fisher. Cuatro de las asociaciones sugieren protección mientras que las restantes tres sugieren riesgo de progresión hacia SIDA. Aún cuando más investigación es necesaria para establecer la relevancia biológica de la asociaciones, nuestra metodología es útil para guiar la búsqueda de hipótesis relevantes y novedosas, así como de describir combinaciones de las características virales con influencia en la progresión clínica.

Table of contents

List of figures

List of tables

List of acronyms

ADA Autism Data Analysis

AI Artificial Intelligence

AIDS Acquired Immune Deficiency Syndrome

AMD Age-Related Macular Degeneration

ANN Artificial Neural Network

APOBEC3 APOlipoprotein B messenger RNA Editing enzyme, Catalytic polypeptide-like

ART AntiRetroviral Therapy

ASD Autistic Spectrum Disorder

BNN Bayesian Neural Network

BPNN Back-propagation Neural Network

BlC Bladder Cancer

BrC Breast Cancer

CENSIDA *Centro Nacional para la Prevención y el Control del VIH y el Sida*

CBFb Core Binding Factor β

CC Colon Cancer

CD4 Cluster of Differentiation 4

CDC Centers for Disease Control and Prevention

CF Confidence Level

CL/P Non-syndromic cleft lip with or without cleft palate

Cul5 Cullin5

CV Cross-Validation

DNA	Deoxyribonucleic acid
EloB	ElonginB
EloC	ElonginC
GA	Genetic Algorithm
GE	Grammatical Evolution
GENN	Grammatical Evolution Artificial Neural Network
GP	Genetic Programming
GPNN	Genetic Programming Artificial Neural Network
GWAS	Genomic Wide Analysis Study
HIV	Human Immunodeficiency Virus
HLA	Human Leucocyte Antigen
ID3	Iterative Dichotomiser 3
IG	Information Gain
KDD	Knowledge Discovery in Databases
KIR	Killer-cell Immunoglobulin-like Receptors
LOAD	Late-Onset Alzheimer's Disease
LR	Logistic Regression
MBS	Multiple Beam Search algorithm
MDR	Multifactor Dimensionality Reduction
MLP	Multi-Layer Perceptron
MM	Multiple Myeloma
MR	Mental Retardation

NLIS	Nuclear Localisation Inhibitory Signal
NIST	National Institute of Standards and Technology
PD	Parkinson's Disease
RF	Random Forest
RNA	Ribonucleic Acid
SCZ	Schizophrenia
SNP	Single Nucleotide Polymorphism
SRBCT	Small Round Blue Cell Tumour
SVM	Support Vector Machine
TB	Tuberculosis
UNAIDS	Joint United Nations Programme on HIV/AIDS
vif	viral infectivity factor
VL	Viral Load

Introduction

This chapter introduces the research work and the current difficulties faced in assessing the interaction among genetic variables. It also presents the organisation of this thesis, the justification of the research work, the principal contributions and description of the research methodology.

The Human Immunodeficiency Virus (HIV) epidemic

The HIV epidemic is still a worldwide problem. According to 2018 estimates by the Joint United Nations Programme on HIV/AIDS (UNAIDS) [175], 37.9 million people were living with HIV worldwide. UNAIDS estimated the number of new infections at 1.7 million while the number of Acquired Immune Deficiency Syndrome (AIDS) related deaths was 770,000. UNAIDS also estimated that 24.5 millions of the people living with HIV had access to treatment.

From the UNAIDS global data, it is notable that the number of AIDS-related deaths has been in decline because of an increase in AntiRetroviral Therapy (ART) availability. This decline has been further bolstered by the World Health Organization recommendation for an earlier ART initiation [128]. The first recommendation is that every patient with HIV should start the ART regardless of the Cluster of Differentiation 4 (CD4)+ T cell count [1]. These situations had helped in reducing the number of AIDS-related deaths from a peak of 1.9 million in 2004 to 770,000 in 2018. The number of new HIV infections has also been in steady decline from its highest value in 1996, 3.4 million, to 1.7 million in 2018. However, in recent years the decline has been slower than expected and even negligible in Latin America where it has only seen a decrease of 1% during the same period.

The same problem exists in Mexico. Mexico hosts the third-largest HIV-infected population in the American continent [174]. Mexico has an estimated 230,000 HIV/AIDS infections from the years 1983 to 2017.

[1] Before the recommendation, the ART was delayed until a specific value in the number of CD4+ T cells were reached (between 200-350 cells per microliter).

Figure 1 shows the number of new HIV/AIDS cases and AIDS-related deaths as registered by the *Centro Nacional para la Prevención y el Control del VIH y el Sida* (CENSIDA) from 1990 to 2018. In 2017 there were 4,720 AIDS-related deaths, which shows a slow decline in the last years from its peak at 5,189 in 2008. The number of new AIDS cases has been in steady decline over the past years from its highest values around 8,800 in 1999 and 2006. However, new HIV infections have been increasing in the last years and even setting a new maximum in 2018 at 10,851. The decreasing number of new AIDS cases reflects an early ART initiation as recommended by the Mexican Health Ministry[2], in alignment with the World Health Organization recommendations. The recommendations also establish the necessity for a continuous evaluation of CD4+ T cells and viral loads. Since 2001, CENSIDA is the decentralised Mexican institution in charge of establishing and operating the governmental strategies against the global epidemic. In 2003, changes in this strategy set the efforts at the state level through the *Centro Ambulatorio para la Prevención y Atención del SIDA e Infecciones de Transmisión Sexual*.

Another important fact is the HIV/AIDS prevalence in the Mexican population living in the U.S. According to estimates from the Centers for Disease Control and Prevention (CDC), the U.S. had 1,040,352 million adolescents and adults living with HIV at the end of 2018. The CDC estimated that new HIV infections declined by 15% between 2010 and 2017, from 44,805 to 37,968. This reduction in the new infections has stalled in recent years. However, there are two ethnic groups disproportionately affected: African-Americans and Hispanics/Latinos. According to the U.S. Census Bureau, African-Americans account for 14% of the population but bear 42% (16,055) of new HIV infections. Hispanics account for 27% of new HIV infections (10,255) but represent only 11% of the population [22]. Hispanics also have an HIV prevalence three times higher than the non-Hispanic white population [54]. This is relevant since Hispanics are the fastest-growing U.S. minority. They accounted for almost 50% of the total population growth in the U.S. from 2010 to 2016. Furthermore, Mexicans represent 64% of Hispanics living in the U.S., as estimated for 2018 by the U.S. Census Bureau [176].

Recent breakthroughs increase the hopes for a vaccine or a cure for the HIV/AIDS infection. One example is HIV remission in six patients accomplished by Salgado et al. in 2019 after the allogeneic haematopoietic stem cell transplantation [154]. The authors even mentioned that five out of the six patients in the study achieved undetectable levels of HIV. The study provides further evidence supporting the important immune mechanisms set in motion after the transplantation. However, this is far from being a cure given the cost, morbidity and

[2]https://www.gob.mx/cms/uploads/attachment/file/411867/Gu_aARV_2019_09Noviembre.pdf

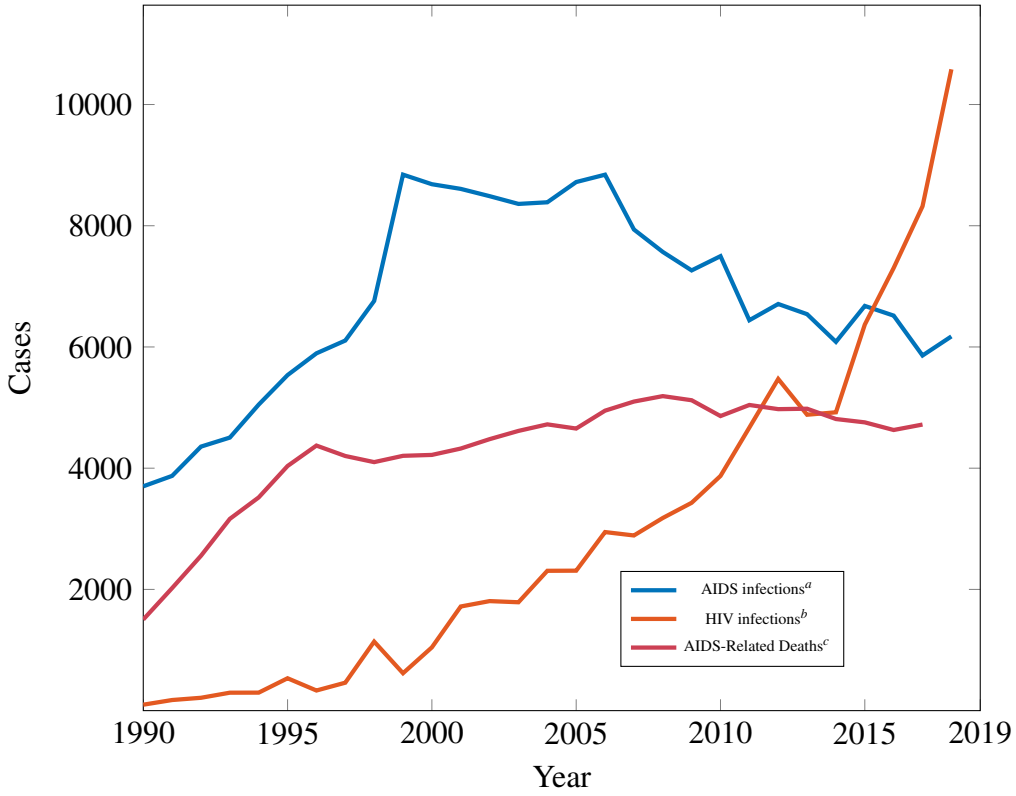


Figure 1. New HIV infections HIV/AIDS, and AIDS-related deaths for Mexico from 1990 to 2018 [23].
[a] Notified AIDS cases by the year of diagnostic.
[b] Notified cases that remain as HIV seropositive by the year of diagnosis.
[c] According to the year of decease.

mortality associated with transplants. Another example is the study by Rodger et al. involving serodiscordant same-sex couples having an ART virally suppressed partner [147] [3]. The results showed a very low infection risk when the patient has a suppressed viral load, supporting the notion that Undetectable corresponds to Untransmittable (U=U) campaign[4]. Nevertheless, achieving an undetectable viral load requires the detection of the HIV infection and correct follow-up of ART adherence. These are problems present even in developed countries, which are further affected by fears, stigma, homophobia, social segregation, access to medical attention, etc. These problems are even bigger in developing countries such as Mexico, where the estimates

[3] Serodiscordant couple refers to partners with distinct HIV serostatus: one tested as positive (with detection of antibodies directed towards HIV) and another tested as negative.
[4] https://www.preventionaccess.org/undetectable

show that only 64% of the people living with HIV know their serostatus, 62% is in ART and only 43% have achieved viral suppression [174].

Some problems associated with HIV control arise from a high mutation rate of the virus that affects the ART-drug resistance and potential for escaping the immune system defences [85, 177]. This high mutation rate influences the HIV genome configuration present in every patient. The HIV genomic configuration has been of particular interest since the discovery of the virus and the scientific community has been achieving important advances in recent years. However, understanding the factors that pinpoint transitions from HIV to AIDS remains elusive, partially because of the great genetic diversity of HIV [50, 122]. Therefore, the study of relevant changes in the virus genetic configuration might help advance the knowledge for the development of a cure or guide preventative strategies. It is even more relevant to study HIV's genetic configuration and its impact on less studied local populations [26, 133].

Researching for genetic associations

Research using genetic information presents the possibility of uncovering some new insights into disease mechanisms. This is a goal of systems biology, along with functional genomics [123] [5] and physiology analysis [70] [6]. System biology aims to provide the possibility of personalised medicine. The individualised treatment would in principle rely on personal detailed genomic information to provide therapeutic interventions that are more precise while diminishing the secondary effects on healthy cells (contrary to chemotherapy, which is a direct attack to all human cells). The relevancy comes from the belief that complex diseases surge from the interaction of three signals: genetic, environmental and pathogen challenges [145, 118]. These variables used for functional genomics data can aid for early disease diagnosis, prognosis and therapy guidance [4]. In addition, they can advance our understanding of pathogenic mechanisms and drug development [173].

Researchers use statistical methods in the search for associations between genetic data and clinical endpoints. Some of the methods more used in the medical field for assessing possible associations are the χ^2 and the Fisher's exact test. Since the tests need hypotheses formulation, the process usually requires the involvement of experts. Their knowledge guides the hypotheses formulation by including relevant combinations based on biological information. However, some limitations on this approach are the current knowledge of biological interactions and the restrictive significance threshold arising from corrections for multiple testing [125, 72]. This

[5] Human Genome Project: https://www.genome.gov/human-genome-project

[6] Human Physiome Project: http://physiomeproject.org/

restriction makes the use of traditional statistical methods more difficult and less useful [118]. Additional obstacles are the high dimensionality of the data [110] and the limitation in the number of samples [43]. The former refers to the high number of variables to investigate. The latter is because of restrictions for acquiring more samples and limitations on the availability of clinical information.

Machine learning approaches in the search for genetic associations

In recent years there has been an increase in using different Artificial Intelligence (AI) methods for the detection of biological patterns and clinical endpoint associations [86, 183, 7]. Most of the methods that researchers have been using for association assessment involve two AI sub-fields: machine learning and pattern recognition. Machine learning studies distinct types of learning paradigms such as supervised learning, unsupervised learning and reinforcement learning [107]. It involves the use of different mathematical models and approaches to enable automatic computer learning based on the data (inductive learning). Pattern recognition is a scientific discipline related to methods for a better object description that would allow a better classification (better discrimination among distinct objects) [102]. The steps involved in pattern recognition include pattern acquisition, feature definition and classifier design or selection [11]. Usually, it is in the last step where some of the machine learning approaches are applied for classification.

Machine learning methods have the advantages of being parameter-free models with the capability of detecting complex non-linear interactions [75, 115, 6]. Some of the most important machine learning approaches include Artificial Neural Network (ANN) [170, 169, 16], Support Vector Machine (SVM) [32], Bayes methods [62, 76], C4.5 [139] and Multifactor Dimensionality Reduction (MDR) [144], among others. ANNs are universal function approximators and a state-of-the-art classifier. Nevertheless, determining the most important input features (e.g. the *genetic variables* that determine a *clinical status*) from the trained models is no easy task. For this reason, their process is considered as a "black box", which is also the case for other approaches such as the SVM. On the contrary, methods such as Iterative Dichotomiser 3 (ID3), C4.5 and MDR include some form of feature assessment in their process. This is also the case for the Apriori algorithm, although with an unsupervised learning approach. Since our interest is identifying biological factors that might have an association with some clinical status, this thesis focus on the approaches Association Rules (Apriori algorithm [1]), Decision Tree Induction (ID3 [138] and C4.5 [139]) and Pattern Mining (MDR [144]). These methods have been applied

separately in previous works; however, this thesis combines their learning capabilities. The reason for using the combination of machine learning approaches is that each of them detects patterns from the data in a distinct way. By combining them, there exists a higher probability of detecting the most relevant patterns while reducing their disadvantages. In this way, the combination of distinct methods can improve disease or prognosis identification [16, 129]. For these reasons, we proposed a machine learning-based methodology for feature assessment and the search of their interactions by considering the patterns detected in the data. After assessing the relevance of the features, the last step in the methodology delivers specific combinations for association testing. We applied our proposed methodology to a new dataset generated from a previous study by Guerra-Palomares et al. [56]. This data describes the HIV genetic configuration sampled from patients with different clinical statuses. Our methodology allowed the suggestion of seven significant associations, four of them indicating protection.

Definitions

Next are presented some relevant definitions.

- Feature / Variable / Attribute: A descriptor of interest with values indicating its presence or levels. For example, the descriptor can describe the specific segment of a genetic sequence.
- Feature Interaction: The combination of two or more variables used for assessing the effect on the information when they are considered together. For example, the features $Var1$ and $Var2$.
- Feature Combination: The specific combination of features with specific values, for example, the features $Var1$ with value 1, $Var2$ with value 0 ($Var1 = 1, Var2 = 0$) which can be used as a hypothesis in a statistical test. These combinations can represent specific genetic patterns.
- Association: A feature combination that presents a statistically significant association between rows and columns after the application of a specific statistical test.

Research question

- What machine learning algorithms for feature extraction can identify genetic combinations of variables associated with clinical endpoints (HIV) in a way that correlates with

traditional statistical methods used by geneticist and molecular biologist in molecular epidemiology?

Research Objectives

Main Objective

To identify clinically relevant associations of viral genetic traits and clinical endpoints in a Mexican mestizo HIV-infected cohort through machine learning approaches for pattern recognition.

Secondary Objectives

- To assess the performance of different machine learning approaches for pattern recognition at classifying patients based on HIV-viral infectivity factor (vif) genetic features (HIV-vif dataset) into clinical endpoint strata such as (CD4+ T cell level) and viral loads (Viral Load (VL)).
- To compare the performance of machine learning approaches versus traditional statistical methods for identifying associations between human and viral genetic traits and clinical endpoints.
- To propose a pattern recognition methodology for the identification of associations of genetic traits and clinical endpoints capable of empowering and guiding traditional statistical approaches.

Research Contributions

- Proposal of a new pattern recognition methodology that could help to predict the infection transition from HIV to AIDS in Mexican mestizo patients.
- The study of machine learning approaches in the search for unprecedented genetic patterns in an original HIV dataset.
- Discovery of new specific genetic combinations for biological testing that could lead to discoveries in the medical field.

Research methodology

This thesis is based on Data Science which has been the focus of significant interest in recent years. There is not a definitive definition for this transdisciplinary emerging science, but according to Dinov: i) it deals with complex and usually unorganised data (raw data), which can be of high volume, from different sources; ii) the main objective is the development of algorithms and methods capable of extracting knowledge from raw data in a semi-automatic or automatic way [35]. As shown in Figure 2, Data Science presents an important intersection among Mathematics/Statistics, Computer Science and Domain Knowledge. This is expressed by the U.S. National Institute of Standards and Technology (NIST) definition of Data Science [124]. Data Science involves knowledge discovery from the data using an inductive learning process (i.e. driven by the data). Combining Mathematics/Statistics with Computer Science has been the base for developing algorithms and methods for inductive learning. Furthermore, combining these approaches with relevant Domain Knowledge allows a more guided search for identifying new insights. In contrast to traditional research by classical statistical methods, the Data Science processes do not need a previous definition or assumption on the model representing the data. Moreover, Data Science does not require a probabilistic sampling design (e.g. random sampling) in contrast to traditional research approaches, where failing in the design can nullify the results [164].

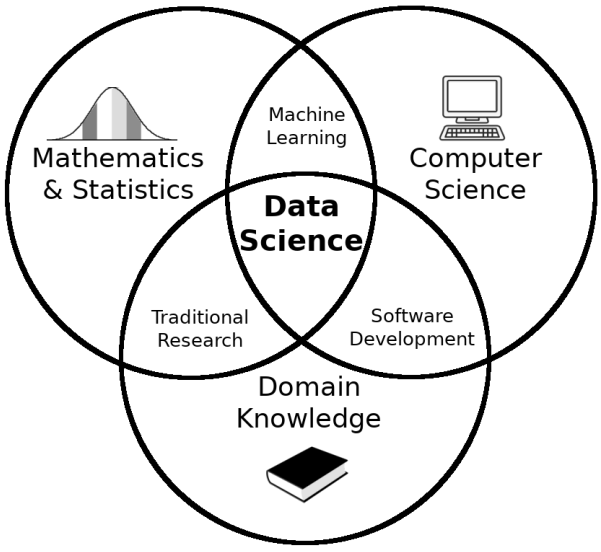


Figure 2. Data Science intersects the fields Mathematical/Statistics, Computer Science and Domain Knowledge (adapted from [124]).

Using a Data Science methodology can give further support to current knowledge while having a higher potential of discovering new insights. Data Science has accelerated knowledge discovery and created a revolution in many scientific fields. For example, AI-systems have surpassed or had similar performance for eye disease diagnosis [46], and at detecting breast cancer on mammograms [105]. Similarly, another AI-system surpassed human pulmonologists at diagnosing the primary disease in patients, identifying correctly 82% of the cases versus 47% from the specialists [171]. Furthermore, an extensive analysis of the AI-systems on disease diagnosis found that its performance was on par with that of health-professionals [94]. Another example is the development of a system for personalised cancer treatment which correctly showed the best therapy according to the literature [69].

One way for applying Data Science is by following the Knowledge Discovery in Databases (KDD) process, which defines the research methodology followed in this thesis [100], see Figure 3. Discovery of new knowledge from the data is the goal of this methodology which considers that usual data representations have no meaning by themselves. The KDD process makes a series of transformations on the data in order to discover knowledge. Such process starts with a high volume of data which transforms to less quantity of information and, finally, to an even lesser amount of knowledge. The abstraction degree, which reflects the acquisition of higher meaning, increases with each transformation. The discovered knowledge can lead to improvements on tasks such as classification, creation of expert systems, diagnosis, among others.

The KDD methodology includes seven major steps (see in Figure 3):

1. Understanding the domain from where the data proceed and establishing the goals. In this step takes place the identification and definition of the algorithms to use depending on the goals.

2. The selection and creation of the dataset for the process. This step involves identifying the data and defining the features for the learning process.

3. Creation of a revised dataset by analysing the original data. This is a very important step since the dataset's quality can improve the discoveries. This might involve addressing missing values, outliers, data type, computational representation and so on.

4. Transforming the data to increase the possibility of a better performance in the learning process. This step can include the use of methods for dimensionality reduction, feature

transformation and so on. Selection of the specific task takes place and could include classification, regression and clustering.

5. Selecting the algorithms to be used for the data mining process

6. Implementing the selected algorithms

This methodology allows returning to previous steps if needed, for example by adding additional features to the dataset the process would return and restart from the fourth step.

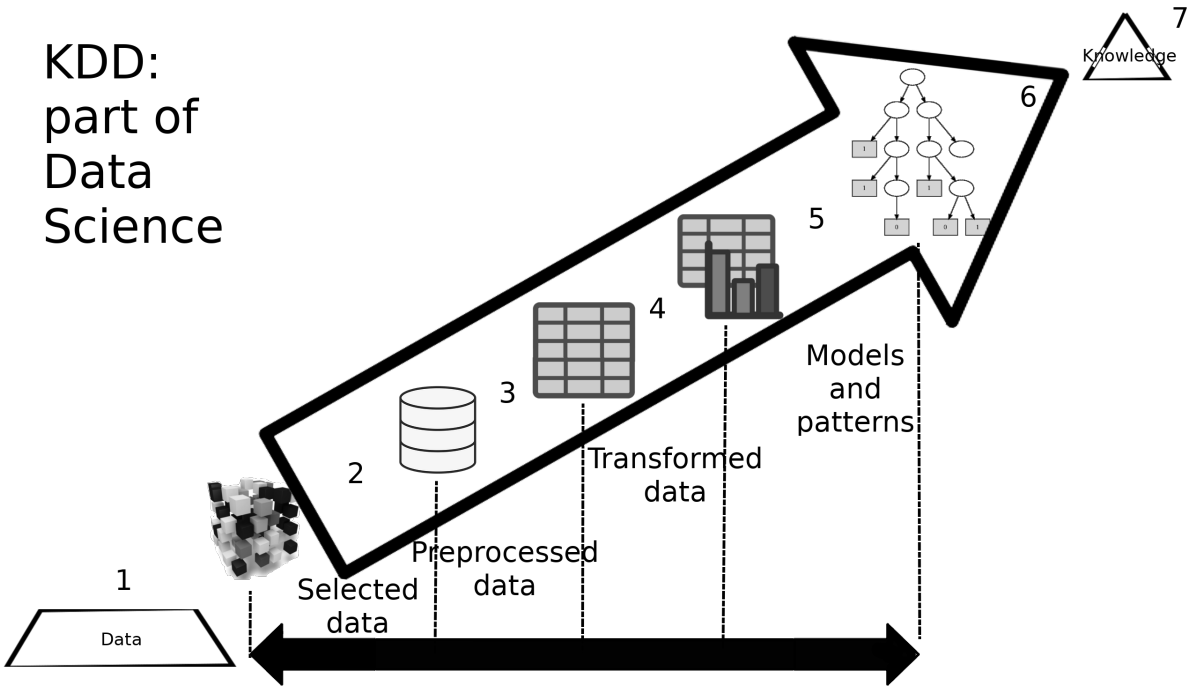


Figure 3. Data Science by the KDD methodology (adapted from [99]). The process starts by applying different transformations on a significant amount of data and finalise identifying new knowledge. The steps in the methodology are: 1) domain understanding and goals; 2) data selection; 3) data cleaning and other tasks; 4) data transformation; 5) data mining; 6) evaluation and interpretation; 7) knowledge discovery.

Publications from this thesis

We have published some results of this thesis. The list in chronological order is below:

- J. S. Altamirano Flores, J. C. Cuevas-Tello, F. E. Martínez Pérez, H. G. Pérez-González, S. E. Nava-Muñoz, "A survey of artificial intelligence algorithms in the search of genetic patterns". In Avances Recientes en Ciencias Computacionales - CiComp 2016 (pp. 233–240). Ensenada, B.C., Mexico: CreateSpace, 2016.

- J. S. Altamirano-Flores, J. C. Cuevas-Tello, and C. A. Garcia-Sepulveda, "Aplicación de aprendizaje autómatico en la búsqueda de asociaciones inmunogenéticas en VIH/SIDA," in Avances en Circuitos y Sistemas, vol. I, Mexico: Universidad Autónoma de San Luis Potosí, 2019, pp. 64–65, isbn:9786075351193.

- J. S. Altamirano-Flores, S. E. Guerra-Palomares, P. G. Hernandez-Sanchez,J. L. Ramirez-Garcialuna, J. R. Arguello-Astorga, D. E. Noyola, J. C.Cuevas-Tello, and C. A. Garcia-Sepulveda, "Identification of HIV-1 vif protein attributes associated with CD4 t cell numbers and viral loads using artificial intelligence algorithms," IEEE Access, vol. 8, pp. 87 214–87 227, 2020.doi: 10.1109/access.2020.2992240.

Thesis organisation

Chapter 1 introduces the biological terms and the dataset related to this work. Chapter 2 presents the machine learning methods that have been used in the search of genetic associations. We describe the proposed methodology for pattern recognition and variable assessment on genetic data in Chapter 3. Results and discussion of the findings are shown in Chapter 4. Finally, the Conclusions presents the findings of this thesis and future work.

Chapter 1

Human immune system and the Human Immunodeficiency Virus

After introducing the relevance on the Human Immunodeficiency Virus (HIV)'s pandemic and the difficulties in the identification of genetic features that might help in predicting the infection progression in the Introduction, this chapter makes some brief biological descriptions. The chapter describes the viruses, HIV and its genetic organisation. It also describes the human immune system, focusing on the APOlipoprotein B messenger RNA Editing enzyme, Catalytic polypeptide-like (APOBEC3) genes and Cluster of Differentiation 4 (CD4)+ T cells. The description of some of their interactions with HIV is also included. Finally, it describes the dataset used in this work and a brief description on the analysis of combinations for hypotheses testing, highlighting the difficulties for this process.

1.1 Genomes

Deoxyribonucleic acid (DNA) of animals and plants comprises a double helix of nucleotides, each composed of a 5 carbon sugar (pentose), an orthophosphoric group and one of four nitrogen bases [adenine (A), cytosine (C), guanine (G) and thymine (T)]. The DNA sequence determines the inherited information for cell function, specialisation and behaviour. The cell copy DNA to identical DNA molecules through the process of replication. To convey the genetic information encoded in the DNA to proteins, DNA is first converted to Ribonucleic Acid (RNA) through a process known as transcription. In a next step, called translation, different amino acids are encoded from the RNA sequences as triplets of nucleotides. Viral genomes can use double-helix

DNA such as that used by animals and plants but may also make use of double-stranded RNA molecules (as happens with HIV), or single DNA or single RNA molecules, encoded in either a positive or a negative sense.

On some occasions, mistakes in RNA and DNA replication process can occur and produce changes in the sequence. These are mutations and could be: i) replacement of one nucleotide for other; ii) multiple deletions (absence of nucleotides on some positions); iii) repetition of some nucleotides added to the sequence or even the translocation of complete segments. Sometimes these mutations have important consequences on a living being. For example, a single substitution on the haemoglobin gene, which encodes the protein for carrying oxygen on the red blood cells, provokes the sickle cell disease, an anaemia type.

1.2 Human immune system

Immunity is an important aspect of recent multicellular organisms as shown by phylogenetic information. The immunity development helps in the protection and defence of the organism against parasitic pathogens such as viruses, bacteria, fungi and protozoa. This protection process includes the use of barriers and pathogen pattern detection, neutralisation and eradication. In the human's case, as in other animals, the immune system has two categories: innate and adaptive. The innate immune system is mainly responsible for a first fast reaction. The reaction normally involves inflammatory responses based on identifying specific molecules on the cell's surface or pathogen. These molecules can be from a microorganism or the result of the activation of non-normal cell processes such as cancer. The recognition process works by using interactions between specific receptors and ligands[1]. Responses from the innate immune system can be fast as it relies on trained defences.

The adaptive immune system takes time (from hours to days) to install a defensive reaction as it involves molecular adaptation to the specific threat. While slower, the adaptive immune defence is normally more specific and effective. The adaptive immune system operates by clonal recognition of antigens which triggers antigen-specific cells generation and a programmed reaction escalation. Clonal recognition involves the use of antigens, a special Y-shaped protein produced mainly by plasma B cells. The cells type B and T represent specialised white cells produced by the bone marrow belonging to the lymphocyte cell line. Most white cells go through a specialisation such as the T cells in the thymus, while a minor proportion specialises

[1]The interaction between receptors and ligands on the cell surface is used by the immune system for the identification of threats for the cells.

as B cells in the bone marrow. B cells produce antibodies which generate immunological memory, perform antigen presentation and help in regulating the cytokine production. Three distinct T cells types exist: i) cytotoxic T cells that kill infected cells; ii) helper T cells that help guide the immune response of both B and other T cells and; iii) regulatory T cells that help in governing the immune system appropriate response. The B cells and T cells belong to the adaptive immune system while the single most important lymphocyte of the innate immune system is the Natural Killer cell.

1.2.1 CD4+ T cells

The CD4 cells or "helper" cells are a subset of thymus-derived lymphocytes (T-lymphocytes) involved in the identification of other cells and harmful agents [96]. The threats identified by the CD4+ T cells get targeted for their destruction by other specific cells such as the Natural Killer cell.

1.2.2 APOBEC3

The human body has many defences against viruses and other threats. One of the most important is the APOBEC3 protein, which is part of the cell's intrinsic immune system [83]. APOBEC3 has known antiviral effects and acts against infections caused by HIV, hepatitis C virus, hepatitis B virus and retrotransposon [24, 56]. The APOBEC3 proteins are coded by seven genes tandemly arranged on the human chromosome 22 [53]. The medical literature identifies these proteins with the letters from A to H, and by far, the best-known is APOBEC3G. APOBEC3G acts against the HIV infection by being included in the virions and impair their function [157]. This happens when HIV's viral infectivity factor (vif) protein is absent. However, when vif is present, it targets the APOBEC3 protein for degradation (i.e. elimination from the body) by hijacking a ubiquitin E3 ligase. The vif protein does this by associating an ElonginB (EloB)-ElonginC (EloC)-Cullin5 (Cul5) E3 ligase complex to APOBEC3G [163]. In this way, HIV provokes the removal of APOBEC3 by using the cell's regulation mechanisms. The mechanism used by vif involves proteosomes which degrade unneeded or misfolded proteins. The proteosomes guide the degradation process by using ubiquitin proteins attached to the targets. In this way, the protein marked for degradation enters a process catalysed by the ubiquitin ligases. Furthermore, vif can also interfere with APOBEC3G's translation process to diminish its presence in the cells [57]. A similar process happens with the other proteins APOBEC3C/D/F/H.

1.3 Human Immunodeficiency Virus

Viruses are considered non-living parasites since they can neither grow nor reproduce on their own. For this reason, the viruses invade host cells and override their processes for surviving and replication. Viruses commonly have a lipid or protein coat containing their genetic material and protecting them from the environment. The coat also contains proteins allowing the virus attachment on specific host cells. This involves the interaction with membrane receptors in the cells that allow access for the viral material. In that respect, we call the entire virus particle capable of infection a virion. Notably, each virus can affect or specialised on a limited set of susceptible cells. For example, HIV targets some cells from the immune system (CD4+ T cells), glial cells and neurons in the central nervous system.

The retroviruses are viruses that infect animals and have a single-strand genome encoded in RNA. Once the virus has penetrated the cell a process starts by the reverse transcriptase which is a viral enzyme. After converting viral RNA into double-stranded DNA, this integrates into the cell's DNA acting as another gene. This allows the viral DNA, called a provirus, transcription using the cell's system for generating viral proteins or for their inclusion in newly generated virions for later release through the cell's membrane. Once new virions get liberated, they restart all the processes on other target cells.

HIV is a retrovirus of the subfamily Lentivirinae characterised by a slowly progressive and indolent infection. The HIV presents a double-membrane envelope having a cone-shaped capsid. This encapsulates a nucleocapsid containing the genomic complex. The typical virion has between 90–130 nm in diameter [90, 113].

Figure 1.1 shows the complete HIV genome, which has 9,719 base pairs as established in the HxB2 HIV-1 sequence reference genome (GenBank: K03455.1). Like other viruses in the Lentivirinae family, HIV has three main genes defining the main proteins: gag for the core polypeptides, pol for reverse transcriptase, protease and integrase, while the envelope coat proteins depend on the env gene. HIV can translate additional accessory proteins that have a great impact on the host's immune system [165]. These proteins are vif, Viral Protein R (vpr), Viral Protein U (vpu), and Negative Regulatory Factor (nef). These genes code for proteins involved in modifying the host cell to enhance virus growth and regulating gene expression [78]. It is thought that some segments in the virus genetic sequence are highly conserved during HIV infection and its transition to Acquired Immune Deficiency Syndrome (AIDS) [108].

As described by Weiss, HIV infection causes AIDS [178]. There are two types of HIV infection: HIV type 1 (HIV-1) and type 2 (HIV-2). These types share only 40-50% of genetic

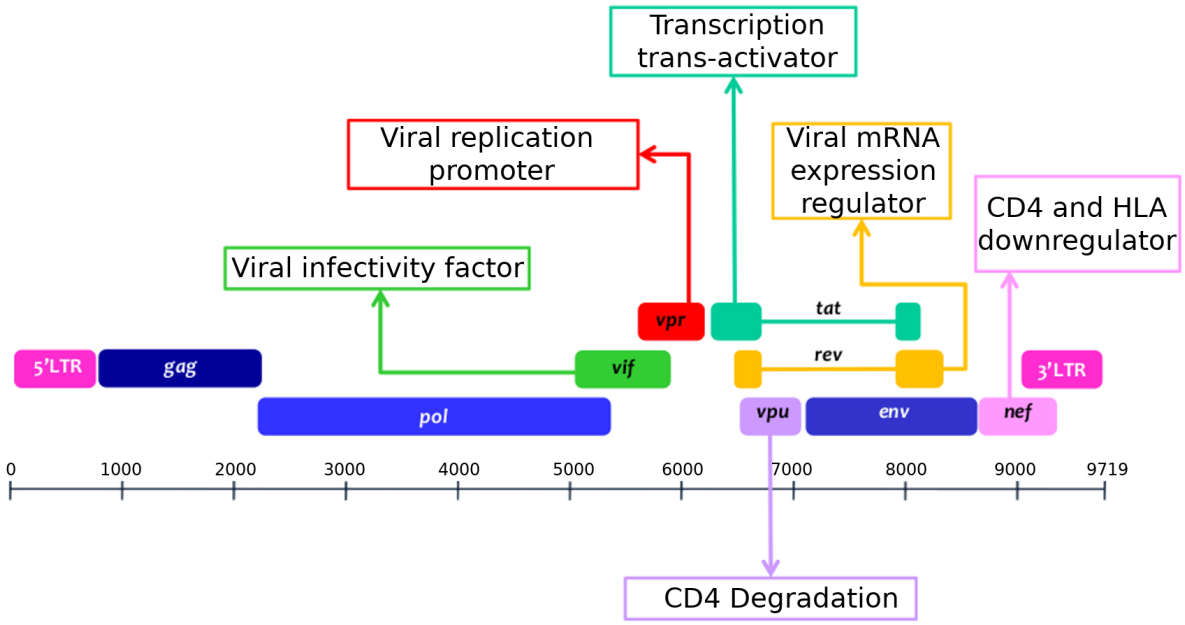


Figure 1.1 Graphical representation of the HIV as established in the HxB2 consensus sequence reference. It includes HIV's three main genes defining its main proteins: *gag*, *pol* and *env*. The vif and the other accessory genes translate proteins that influence modification on the host cells, enhancing virus growth and viral gene expression (adapted from [55]).

homology. HIV-1 is the most common infection in humans, has higher virulence and is thought of having being originated in Africa from cross-species infections [151]. The natural history of untreated HIV infection starts with the transmission, usually through contact with mucosal barriers which are the initial infection site. From there, the virus travels to regional lymph nodes where gets amplified and reproduced. This shows the first stage in the virus phase through a burst in viremia that can reach over 1,000,000 copies per blood millilitre (viral load), shown as the "Acute phase" in Figure 1.2. This phase normally lasts up to three months along with the infection dissemination to the CD4+ T cells, which are the virus major target and one of its latent reservoirs [38]. HIV affects the CD4+ T cells capability to proliferate, to take part in anti-HIV responses, limits cytotoxicity against HIV-infected cells and, secondarily, affects the maintenance of a T lymphocyte subpopulation stable memory [15, 130, 126]. This is because of the HIV's cellular receptor high affinity with the CD4+ antigen present on T-cells, monocytes, macrophages, microglia and dendritic cells. In this pre-seroconversion phase, the patients can present "flu-like" or "mononucleosis-like" symptoms with fever, fatigue, and rash.

The initial viral burst ensures an immune response characterised by creating virus-specific cytotoxic T cells and neutralising antibodies. This response results in a viral "set point". The number of virus copies and CD4+ T cells at this point serves as indicators for the infection prognosis. Individuals with higher viral loads generally transit more rapidly to AIDS with faster diminishing in the CD4+ T cells. The patients' ability to generate and sustain a cytotoxic T-cell response also plays a part in transiting from HIV to AIDS. In this sense, at least three distinct groups exist as shown in Figure 1.2: i) the typical or average progressors, ii) the long-term non-progressors, and iii) the rapid progressors. In the typical HIV infection, the chronic asymptomatic phase can span from 3 to 20 years before AIDS. On the contrary, the rapid progressors typically advance to the symptomatic phase (AIDS) within 3-5 years. The long-term non-progressors represent less than 1% of the HIV patients and can remain healthy for lengthy periods even without AntiRetroviral Therapy (ART). These patients can control the infection for well over a decade sustaining nonsignificant descend on CD4+ T cells or high viral load. Long-term non-progressors can be further divided into elite controllers, patients who can achieve undetectable levels of viral load (<50 copies/ml) for an unspecified period [135].

1.4 HIV-vif Dataset description

The research on HIV has allowed the identification of genetic polymorphisms (mutations present in over 1.0% of the population) that have major effects on its progression. One example is the discovery that being homozygous for a 32 base-pair deletion in the CCR5 gene provides resistance to the infection [93, 85]. Nevertheless, HIV exhibits a great genetic diversity, allowing it to escape the immune system through mutation and an error-prone replication process [162]. For these reasons, it is still of vital importance finding mutations that: i) can provide resistance to the virus, ii) identify those that allow it to improve its infectivity, iii) allow to escaping immune responses. And although research has been leading to important advances in the infection's knowledge in recent years, understanding the factors that pinpoint transiting from HIV to AIDS remains elusive [50, 122]. Searching for the relevant polymorphism in the human genome is one way to face this task, for example, those in the Human Leucocyte Antigen (HLA), Killer-cell Immunoglobulin-like Receptors (KIR) or APOBEC3 genes. Another possibility is to search for mutations in the virus genome and their association with the infection progression. A more complex approach would imply the use of both genetic data and include environmental information, two fundamental biological signals, which would help advance to more personalised therapy options [4].

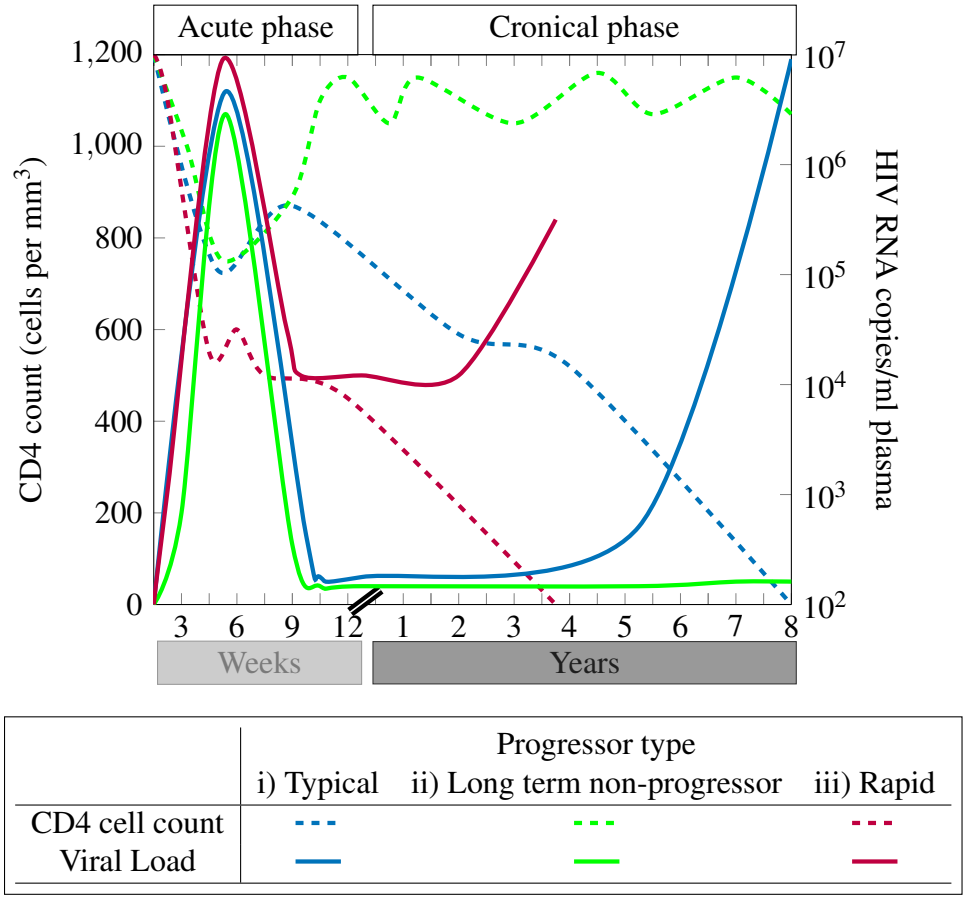
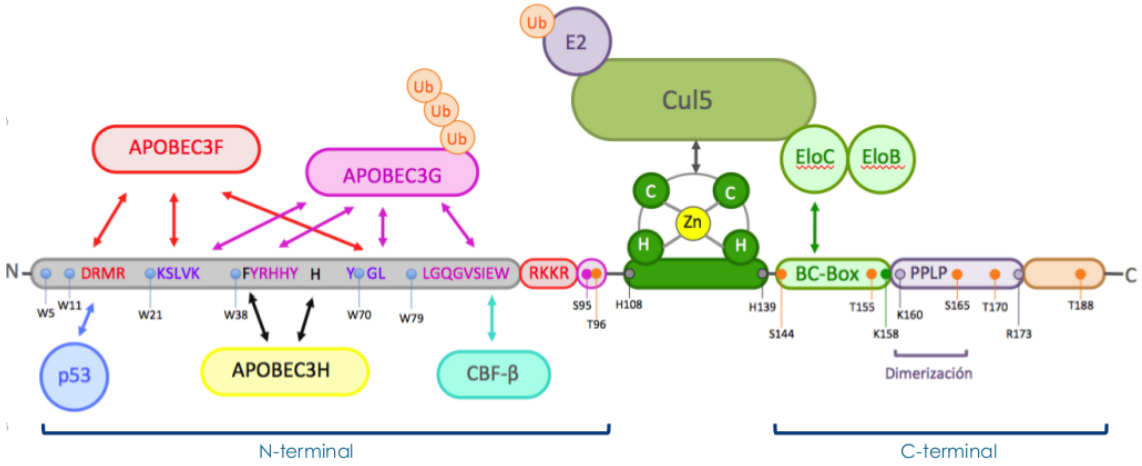


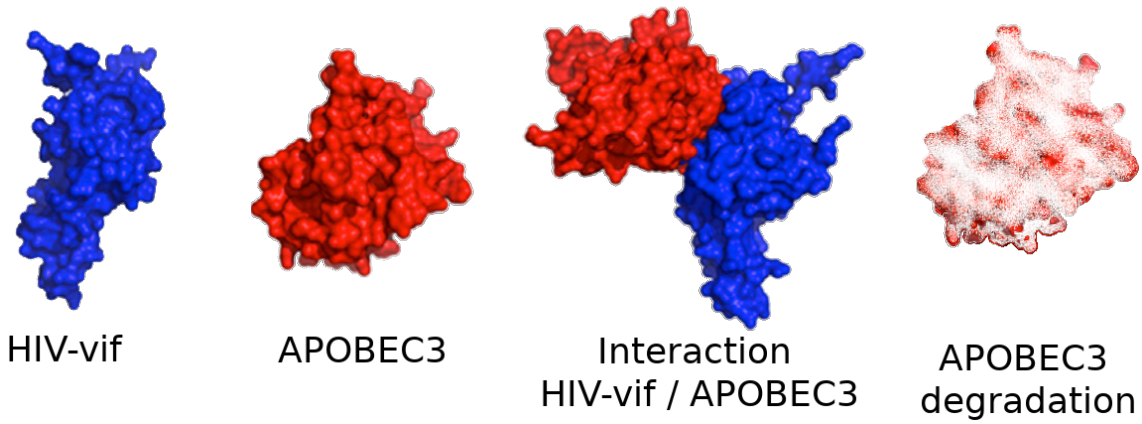
Figure 1.2 Typical HIV to AIDS evolution times (adapted from [135]).

The dataset used in this work comes from analysing viral genetic data in the search for mutations that might associate with a specific clinical status. For this reason, the dataset encodes segments from the vif gene and is hereafter called *HIV-vif*. The vif gene encodes a 192-amino acid HIV accessory protein essential for viral replication [182, 141]. Figure 1.3a shows vif's functional sites. These segments are important since the vif protein counteracts host antiviral proteins of the APOBEC3 family, as mentioned in subsection 1.2.2. The interaction between APOBEC3 and HIV-vif is shown in Figure 1.3b.

The *Laboratorio de Genómica Viral y Humana* from the *Facultad de Medicina* at the *Universidad Autónoma de San Luis Potosí* compiled the genetic dataset. This dataset contains information describing genetic mutations of the viral vif gene. We generated the variables in the dataset from previous research by Guerra-Palomares et al. [56]. The authors in such research processed blood samples from HIV Mexican-mestizo patients to process proviral DNA and



(a)



(b)

Figure 1.3 a) Functional regions of vif (taken from [56]). b) Representation on the interaction HIV-vif and APOBEC3 (adapted from [89]). If there are not relevant mutations, the vif protein causes the APOBEC3 elimination from the cells (adapted from [89]).

then extract viral genetic sequence data for the HIV-vif segment. These patients were receiving clinical attention at the *Centro Ambulatorio para la Prevención y Atención en Sida e Infecciones de Transmisión Sexual* at San Luis Potosí, Mexico. Guerra-Palomares et al. used bioinformatics tools on raw DNA sequence for translating the sequence into the encoded amino acid sequence for each sample to create the original dataset. The dataset comprised 77 sequences from 68 HIV patients and described 192 amino acid positions. As mentioned by the authors, samples were taken on two different occasions from six patients while in three different moments from

another two patients in order to assess the quiasispecies diversification [56]. Figure 1.4 shows the steps followed for the dataset definition for two variables. A further pre-processing allowed representing descriptive variables of functional regions in vif in the HIV-vif dataset. These variables refer to segments important in the vif-APOBEC3 interaction.

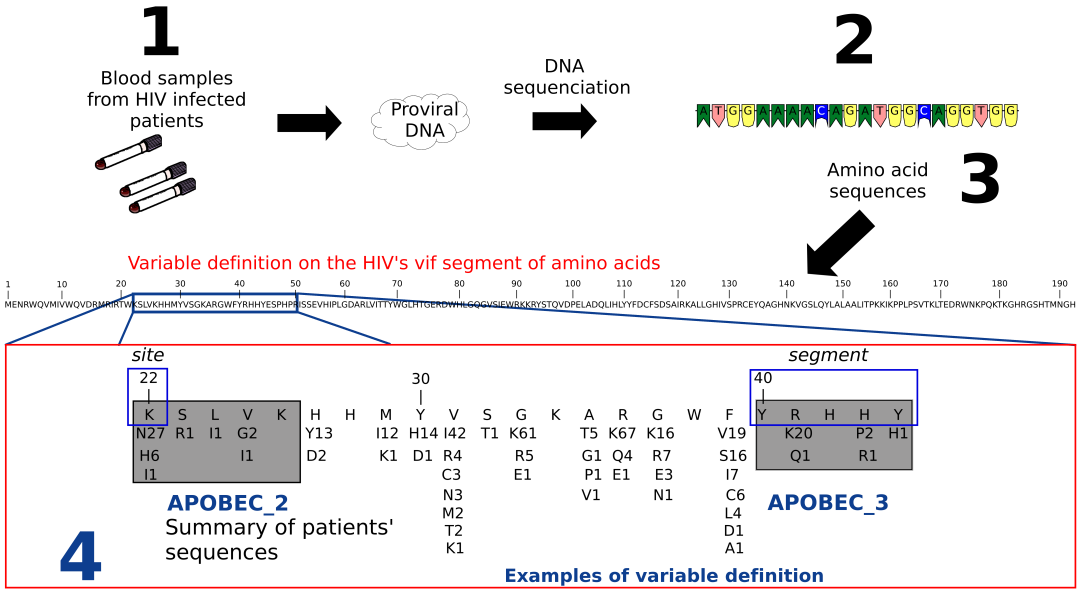


Figure 1.4 Steps for the definition of 17 variables using HIV patients sequences. The first three steps correspond to a previous study by Guerra-Palomares et al. [56], where the amino acid sequences were determined from blood samples after obtaining and sequencing the proviral DNA. The variables were defined in the fourth step using the conservation or chemically relevant mutations on specific segments of the amino acid sequence. For example, the variable "*APOBEC-2*" shows if there exists conservation in the sequence in a specific *segment* (site positions between 22 and 26). Figure 1.5 shows the segments describing each variable. The variables reflect segments important for the interaction vif-APOBEC3.

The sequences from these patients were compared against the HxB2 HIV-1 consensus reference genome for mutation assessment. The original dataset contained 77 vif sequences $(\mathbf{s_1}, \cdots, \mathbf{s_{77}})$ with 192 variables representing the amino acids coded by the vif gene. The original dataset was then modified to represent only seventeen variables, representing the most important sites[2] related to the APOBEC3-vif interaction, see Table 1.1 and Figure 1.5. The definition of the positions/segments was made by the *Laboratorio de Genómica Viral y Humana* for an easier results interpretation. In this way, the analysis was based on non-synonymous substitutions

[2]Each position on the genome reference is called a *site* while *segment* describes some combination of sites.

occurring in vif regions and motifs currently known to have a biological role. The process for determining the number of variables and data type is explained in Annexe B.

Table 1.1 Example of the variables and outputs in the dataset.

Sequence	APOBEC-1	APOBEC-2	APOBEC-3	APOBEC-4	APOBEC-5	APOBEC-6	APOBEC-7	APOBEC-8	CBFb-1	CBFb-2	NLIS	Cul5-1	Cul5-2	Cul5-3	BCbox-1	BCbox-2	BCbox-3	CD4Ini	CD4Hist	VLIni	VLHist
s_1	0	1	0	0	0	0	1	0	0	1	0	0	0	1	0	1	0	1	1	1	0
s_2	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	1
s_3	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	1
s_4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1
s_5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1
s_6	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0
s_7	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0
s_8	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	1	0
s_9	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	1	1	1	0
s_{10}	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
s_{68}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0
s_{69}	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	1	1	0	0
s_{70}	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	0
s_{71}	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1
s_{72}	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	1	1	0	0	1	1
s_{73}	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	1	1	1	1
s_{74}	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	1	1	0	0
s_{75}	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
s_{76}	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
s_{77}	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1

Each variable encodes relevant mutations in areas which interact with APOBEC3 (see Figure 1.5). Eight APOBEC variables represent segments related with the interaction of vif with the proteins APOBEC3C/D/F/G/H: APOBEC-1 (segment 14-17), APOBEC-2 (segment 22-26), APOBEC-3 (segment 40-44), APOBEC-4 (sites 39 and 48), APOBEC-5 (segment 69-72), APOBEC-6 (segment 74-78), APOBEC-7 (segment 81-88), APOBEC-8 (segment 171-175). Two variables pertain to vif segments that allow bounding with Core Binding Factor β (CBFb) to maintain stable vif expression levels and virus infectivity [9]: CBFb-1 (segment 84-89),

CBFb-2 (segment 102-107). One variable reflects the changes in the segment corresponding to the Nuclear Localisation Inhibitory Signal (NLIS) (segment 90-93) important for cytoplasmic localisation in the target cell of the vif protein by interfering with the nuclear import pathway. Three variables relate with elongin B/C-box binding motif Bcbox-1 (segment 144-149), Bcbox-2 (segment 151-154), and Bcbox-3 (segment 156-158). The sites relevant for the Cullin binding motif and box: Cul5-1 (segment 120-121), Cul5-2 (site 124), Cul5-3 (segment 160-169). The Bcbox and Cul5 segments are essential to the elongin B/C-box C-Cul5 E3 ligase through which vif promotes APOBEC3G degradation [163].

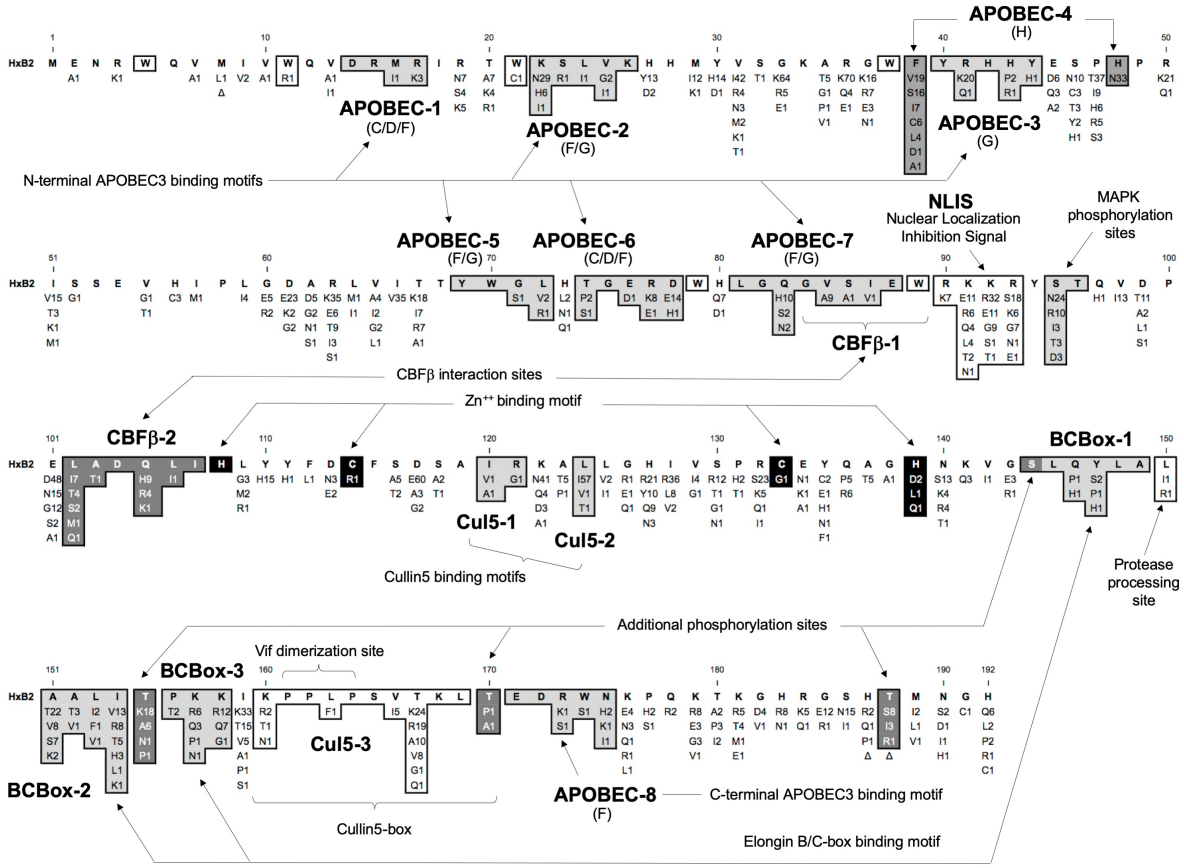


Figure 1.5 Positions from the HxB2 reference for the vif segment used in the seventeen variables definition.

We used a binary representation for all the variables, having a value of one if there was at least one chemically relevant mutation (“mutated” or “mut”). An example of chemically relevant mutation is the change in the position 151 of alanine (A) for serine (S) which describes

substituting one nonpolar amino acid for a polar one represented by the latter. As mentioned before, these changes had a coded value of "1". When samples did not exhibit biologically relevant mutations, they had a coded valued of "0". All the segments are important for their influence inducing APOBEC3 degradation.

The clinical data used for the outputs include measurements of the CD4+ T cells and viral load from the patients. CD4+ T cells are part of the immune system and help to identify infected cells for elimination. Counting CD4+ T cells per microliter (cells/μL) is an important indicator of progression towards AIDS [135]. The HIV viral load refers to the number of viral copies per millilitre in the patients. Initial levels (CD4Ini, VLIni) refers to the first measurement once the infection was detected, while the historical (CD4Hist, VLHist) refers to the median of quarterly assessments during two years.

For the CD4+ T measurement case, the records with $\geq$ 500 cells/μL were codified as "0", otherwise, they were coded as "1". The former can be considered as normal and the latter as an indicator of the infection progression as per international standards [40]. Similarly, the patients with a viral load with less than 10,000 copies per millilitre were codified as "0", otherwise, they were coded as "1". In this case, a value $\geq$10,000 is considered as a high level of viral presence indicating a more advance in the infection [44]. For both the CD4+ T cells and the viral load, a value of "1" indicates an unhealthier status, see Table 1.1. The number of cases for each output and class is shown in Table 1.2.

Table 1.2 Number of sequences per output and class.

			Class	
Sequences	Variables	Output	0	1
77	17	*CD4Ini*	16	61
		CD4Hist	16	61
		VLIni	28	49
		VLHist	44	33

1.5 Analysis on the combinations for hypotheses testing

Hypothesis testing using statistical analysis is one way for searching associations between genetic markers and clinical status. This approach involves applying statistical tests on pre-determined combinations (i.e. hypotheses) generally based on experts' knowledge. However, this process is tedious and time-consuming. On the other hand, this process can also require

exploring a great number of combinations and applying some correction method. Equation 1.1 gives the number of combinations for testing with n binary variables:

$$\sum_{k=1}^{n} 2^k \times \binom{n}{k} \tag{1.1}$$

where k defines the combination size. However, Equation 1.1 must be multiplied by two if the output involves binary values. For example, the number of unique combinations for three binary variables ($n = 3$) is 26, as exemplified in Table 1.3. On the other hand, applying a correction method might require a highly stringent significance level (i.e. α-value), further increasing the difficulties in the search for associations.

Table 1.3 Example of the binary combinations considering three variables. The crossed-out values show that the specific variable-value combination has already been considered. The total number of distinct combinations for three binary variables is 26.

CombId	Variable-values	Comb k=2	Comb k=3	Total Comb
	~~A=0~~	~~A=0, B=0~~		
Comb1		A=0, C=0	A=0, B=0, C=0	
	B=0 ~~C=0~~	B=0, C=0		
Comb1	3	3	1	7
	~~A=0~~	A=0, B=1		
Comb3		~~A=0, C=0~~	A=0, B=1, C=0	
	B=1 ~~C=0~~	B=1, C=0		
Comb3	1	2	1	4
	~~A=1~~	A=1, B=1		
Comb5		~~A=1, C=0~~	A=1, B=1, C=0	
	~~B=1~~ ~~C=0~~	~~B=1, C=0~~		
Comb5	0	1	1	2
	~~A=0~~	~~A=0, B=1~~		
Comb7		~~A=0, C=1~~	A=0, B=1, C=1	
	~~B=1~~ ~~C=1~~	B=1, C=1		
Comb7	0	1	1	2

CombId	Variable-values	Comb k=2	Comb k=3	Total Comb
	A=1	A=1, B=0		
Comb2		A=1, C=0	A=1, B=0, C=0	
	~~B=0~~ ~~C=0~~	~~B=0, C=0~~		
Comb2	1	2	1	4
	~~A=0~~	~~A=0, B=0~~		
Comb4		A=0, C=1	A=0, B=0, C=1	
	~~B=0~~ C=1	B=0, C=1		
Comb4	1	2	1	4
	~~A=1~~	~~A=1, B=0~~		
Comb6		A=1, C=1	A=1, B=0, C=1	
	~~B=0~~ ~~C=1~~	~~B=0, C=1~~		
Comb6	0	1	1	2
	~~A=1~~	~~A=1, B=1~~		
Comb8		~~A=1, C=1~~	A=1, B=1, C=1	
	~~B=1~~ ~~C=1~~	~~B=1, C=1~~		
Comb8	0	0	1	1
			Distinct Combinations	26

This example shows that even with a small set of variables, the number of possibles combinations when considering binary values and outputs results in a combinatorial explosion. And, as mentioned before, this problem is further complicated in the search of associations where exploring all the possible combinations would result in highly stringent significance value. Such a small value would be needed for decreasing the probability of finding associations just at random after testing so many of them.

Chapter 2

Machine learning methods

This chapter presents a review of machine learning methods used for feature assessment, selection and classification. These methods include paradigms such as Artificial Neural Networks (ANNs), Bayesian methods, Decision Tree Induction, kernel methods, Association Rule Mining, among others. A brief introduction to Artificial Intelligence (AI) and machine learning is also given.

2.1 Artificial Intelligence

AI is a computer science field interested in developing machines that might exhibit *intelligent* behaviour. Since there is not a definitive standard definition on what intelligence is [88], to the date there is not a single multipurpose approach that can cover all the possible aspects of intelligence. For this reason, as mentioned by Rusell, AI advancement has been historically guided by four approaches: i) thinking humanly, ii) acting humanly, iii) thinking rationally, and iv) acting rationally. The first two approaches base their goals in relation with humans while the last two are according to achieving the best possible results, independently on how humans do it. There are several definitions of AI, but most of them include two relevant aspects: i) it deals with the human thinking process, and ii) the representation of the human thinking processes in the machines [153]. One definition from Rich & Knight that summarises the goal of AI is: "Artificial Intelligence is the study of how to make computers do things at which, at the moment, people are better" [142]. The intelligent behaviour that AI is trying to capture is present in everyday human activities. Examples of these activities include: trying to determine the best path to follow when transiting from one point to another, the identification and recognition

of objects, planning activities, playing games, etc. The AI community has been researching distinct approaches for its advancement in subfields such as pattern recognition, computer vision, robotics, expert systems and machine learning.

The formal beginning of AI was the 1956 Dartmouth Conference on AI with the participation of important scientists such as Marvin Minsky and Claude Shannon. Alan Turing had an important influence on the rising area through his famous paper "Computing Machinery and Intelligence" from 1950 [172]. However, limits in computational power, data and unrealistic goals produced a great disenchantment after important initial achievements. This was reflected in the years following 1965 when interest in AI diminished. A posterior period of high relevance resurfaced between 1970 and 1980 with the advancements on Expert Systems [153]. These created significant interest in the industry, but limitations on this approach were notable in the second part of the decade of 1980. Research on other approaches led to great advances in the same decade, allowing a resurgence of interest in AI. In this sense, the development of ANN's backpropagation training method in 1986 was one of the most relevant [152]. The 1990s saw the surge of Support Vector Machines (SVMs), which improved the highest performances on distinct problems such as pattern recognition [30]. Meanwhile, some problems limiting improvement on ANNs performance remained until 2006 when the research group led by Geoffrey Hinton developed what is know as Deep Neural Networks [66]. This and other novel approaches have reignited a recent period of significant interest from the industry which has remained. This interest has been further impulsed by the availability of faster computers and fast-increasing data availability. The present synergy has allowed the advancement on applications ranging from autonomous driving to talking digital assistants.

2.1.1 Machine learning

As defined by Mitchell, "the field of machine learning is concerned with the question of how to construct computer programs that automatically improve with experience" [107]. Machine learning is a subfield of AI that relates to the improvement in the performance on a task by learning. Machine learning has three training approaches: supervised, unsupervised and learning by reinforcement as shown in Table 2.1. The supervised approach (see Figure 2.1) requires *labelled* data which can be described considering N examples, where the i^{th} example is a tuple (x_i, y_i). The values in x_i describe the characteristics of the object (inputs), while y_i describes the corresponding response (output or *labels*). Supervised learning can be applied for classification and regression. In classification, the goal is to learn a function that maps from

known inputs to outputs (i.e. the labels representing classes). The regression process is similar, but instead of discrete outputs, the method must approximate numerical values as in function approximation. Unsupervised learning (see Figure 2.2) does not need labelled examples for guiding the process. The most common task in unsupervised learning is clustering. On the other hand, reinforcement learning (see Figure 2.3) is guided by the effects that some actions have on a specific problem. The process considers some policies indicating the steps to follow and the rewards or punishments that will help in the learning process.

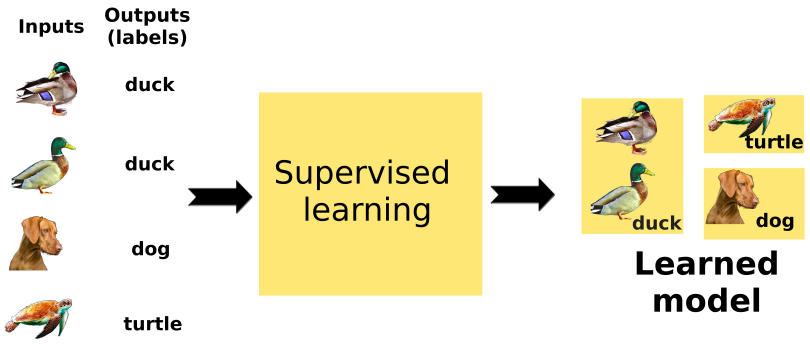
Table 2.1 Distinct machine learning approaches for training.

Approach	Description
Supervised learning	The learning process has information available (i.e. *labels*) for guiding the model creation capable of mapping the relation between some known inputs and known labels. In this way, the process makes an approximation of the underlining function related to the data.
Unsupervised learning	The learning process to recognise patterns without the use of known input-output examples.
Reinforcement learning	The learning process is further incentivised by using a series of "rewards" and "punishments" which can speed up and improve the learning process.

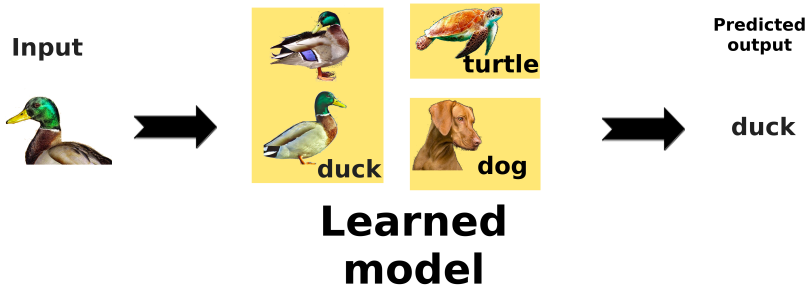
2.1.2 Pattern Recognition and feature selection

As mentioned by Marques De Sá, in the AI field, the data that describes an object is a *"pattern"* and the scientific discipline dealing with this data and object classification is Pattern Recognition [102]. Some AI applications mentioned by Theodoris include artificial vision, text recognition (words or numbers), computer-assisted disease diagnosis, speech recognition, data mining and knowledge discovery, among others [168]. Pattern recognition is a valuable tool since the exact mathematical modelling from the data is a hard task in many problems, if not impossible with the current knowledge.

Another important area that has been getting increased attention is feature selection, which helps in dimensionality reduction and in identifying the most important characteristics from some data. Dimensionality reduction deals with identifying the k most relevant features from a bigger set with n features present in some dataset. The goal is that the smaller set of k factors can represent the most relevant information present in the data. Dimensionality reduction normally



(a)



(b)

Figure 2.1 Example of a supervised learning process for classification considering two phases. a) In the training process, input data and its known output (i.e. *labels* or *targets*, called *class* when the task is classification) values are entered in a specific method. This generates the learned model, a mapping between the input and output values. b) For predicting, other examples can be processed as input on the learned model, which returns the label specified by the model.

involves one of the following approaches: filtering, wrapping and embedding. The filtering approach measures the training data features by ranking their information in the identification of some predictor. Once the features have a rank, those capturing most of the information get selected as the most relevant. The wrapping approach makes use of machine learning approaches combined with feature subsets for assessing the impact of the features on the classification performance. In this way, the features in the subsets that had a higher classification performance get selected as the most relevant. The embedded approach includes the feature assessment task as part of a machine learning process for model creation [16].

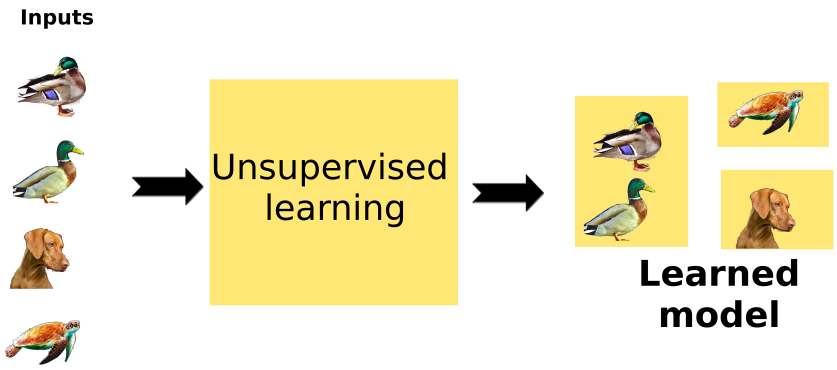


Figure 2.2 Example of an unsupervised learning process for clustering. In this case, the input data has no specific output values. The clustering process identifies elements with similar characteristics and generated clusters according to them.

2.1.3 History of AI applications in medicine

The Expert Systems are examples of successful applications on the use of AI in medicine which helped to reignite the interest in the area. Some of the most famous systems appeared in the 1970s, such as MACSYMA developed at the Massachusetts Institute of Technology [103]. Other examples include MYCIN [160] and DENDRAL [20], developed at Stanford University. From these, MYCIN was the first applied in medicine and helped in the diagnosis and treatment of bacterial, virus or fungal infections based on reported symptoms and clinical information [19]. These systems based their functions on the use of rules and inferences processes. However,

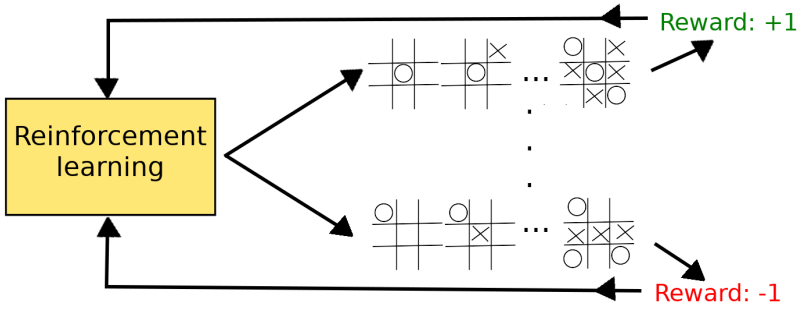


Figure 2.3 Reinforcement learning involves assessing a specific environment and the application of actions according to some policy. It also involves an evaluation of the performance according to some goal. According to the improvement or worsen in the performance, a reward or a punishment is given and a policy update takes place.

acquiring the knowledge from the experts and finding a suitable representation can be a laborious process. This task required the determination of how the experts resolve problems and its translation to an appropriate machine representation. The major difficulty lies in the hardship for gathering the relevant knowledge from the human experts, which is a complex process. And even when finding new discoveries is not its purpose, it is highly probable that little to none new knowledge might arise from its execution.

Recent advancements in machine learning approaches allow new possibilities in the search of knowledge discovery. The approaches of machine learning have the advantages of being model-free[1] and capable of detecting complex non-linear interactions[2] [75, 115, 6]. For these and other reasons, machine learning has been attracting attention in medicine. As mentioned by Becker, some major uses of AI in medicine include disease risk assessment, management, aetiological research, treatment recommendation and patient assistance [7]. According to Bisaso et al., machine learning use in Human Immunodeficiency Virus (HIV) research had three stages in the period from 1995 to 2017 [13]. The first stage, between 1995 and 2002, comprised the use of data from clinical information. A second stage, between 2003 and 2015, included an increase in using genetic information because of advances in genotyping technologies. Finally, the third stage, from 2016 to 2017, shows an important increase in imaging data opening new lines of research. The objectives for the first two stages mainly focused on prediction using mostly ANNs and Logistic Regression (LR). The second stage objectives also focused on analysing the association between genetic characteristics and clinical events/AntiRetroviral Therapy (ART) resistance. This has also been the case for the third stage, along with the association between imaging data and clinical events. The research using machine learning in HIV mostly has related to classification algorithms like SVM, Random Forest (RF), Decision Trees, or through the use of ensembles[3].

[1] A model-based approach is based on statistical a priori considerations among the variables and model assumptions on the probability distribution. For example, for the model to have a satisfactory performance the data must be represented by a specific probability distribution or must have independent variables. Since this can not be always satisfied, these methods are not suitable for some problems. On the other hand, the model-free approaches do not rely on a priori satisfaction of conditions [52].

[2] A linear interaction is one that can be explained by an additive model in the form $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$. This model explains the values of y by adding a constant value (β_0) to the product of coefficients ($\beta_1 \ldots \beta_n$) and explaining variables ($x_1 \ldots x_n$). The linear model is more suited for explaining data where the variables by themselves highly influence the value of y (*main effects*). On the other hand, this model is not suitable for more complex data where the interactions among the variables have a great influence on the value of y. It is thought that this is the case for complex disease risk, which is considered to arise from a combination of interactions between genetic and environmental signs [145, 118, 116].

[3] An ensemble is the combination of some machine learning methods that are trained in a problem. Each of these methods learns a model from the problem and then can generate prediction by themselves. Then a final

2.2 Literature review

The next subsections present some machine learning methods that have been used in the search of associations between genetic traits and illnesses. The methods are grouped by the main approach used in each work. Table 2.2 shows some of the diseases studied by using machine learning approaches either on Single Nucleotide Polymorphisms (SNPs), Genomic Wide Analysis Study (GWAS), clinical information or other data sources. Tables 2.3 and 2.4 show the summaries on the use of these approaches on real and simulated data. From these, the more related with this work are those applying a hybrid approach since they combine different machine learning methods.

Table 2.2 Some of the diseases studied with machine learning approaches.

Disease	Study
Autism Data Analysis (ADA)	[62]
Age-Related Macular Degeneration (AMD)	[62]
Autistic Spectrum Disorder (ASD)	[16]
Breast Cancer (BrC)	[144, 16]
Bladder Cancer (BlC)	[91]
Colon Cancer (CC)	[16]
Non-syndromic cleft lip with or without cleft palate (CL/P)	[92]
HIV	[116, 173]
Leukaemia	[148, 170, 32]
Late-Onset Alzheimer's Disease (LOAD)	[62, 76]
Multiple Myeloma (MM)	[63]
Mental Retardation (MR)	[16]
Parkinson's Disease (PD)	[117]
Schizophrenia (SCZ)	[72]
Small Round Blue Cell Tumour (SRBCT)	[79, 170, 169]
Tuberculosis (TB)	[6]

2.2.1 Artificial Neural Networks

As mentioned by Jain and Mohiuddin, ANNs can model complex non-linear systems and can solve a great number of problems such as pattern recognition, prediction and function

prediction is generated by summarising the decisions made from every individually model. For example, in classification, the final decision would be given by the class with more "votes" (i.e. the class more frequently predicted) from the learned models.

Table 2.3 Works revised for the literature review per approach used and disease on real data, considering their number of variables.

Approach	Disease	SNP/Vars.	Cases	Controls	Ref.
ANNs	HIV	137	867		[116]
	PD	70	305	321	[117]
	SRBCT	2380	63		[79]
	TB	16925	104		[6]
Bayesian probability	ADA	8198	1783	2439	[62]
	AMD	51	96	50	[62]
	LOAD	287479	857	551	[62]
		312260	299	290	[76]
Data mining approaches	BrC	10	200	200	[144]
	Leukaemia	17	43	300	[148]
Hybrids approaches	ASD	+250K	567	567	[16]
	BrC	+500K	111	111	[16]
	CC	+250K	42	42	[16]
	Leukaemia	7129	72		[170]
	MM	12000	218	45	[63]
		+250K	360	360	[16]
	SRBCT	2380	63		[170]
		96	63		[169]
LR	CL/P	741	1576		[92]
	SCZ	86	3063	2847	[72]
SVMs	Leukaemia	23	43	300	[32]
Tree based approaches	BlC	7	354	560	[91]
		39	491	791	[91]
	CL/P	741	1576		[92]
	HIV	88	75	21	[173]

Table 2.4 Works revised for the literature review per approach used and disease on artificial data, considering their number of variables.

Disease	Approach	Datasets	SNPs/Vars.	Cases	Controls	Ref.
Simulated data	ANN	25	8 or 20	200	200	[145]
		12,000	20-24	200, 400 or 800	200, 400 or 800	[117]
		2,000	10	200	200	[114]
		5	10	200	200	[119]
		3,000	100	500	500	[118]
		12	6	100	100	[58]
		1,200 and 6,000	50 or 1,000	1,000	1,000	[6]
	Bayesian	50	100	1,000	1,000	[62]
		100	1,000	1,000	1,000	[76]
	Data mining	200	10	200	200	[144]
	Hybrids	-	25,000	100		[169]
	SVM	-	23	43	300	[32]
	Tree based	16,000	5,10,15,20,25	2,000 or 4,000	2,000 or 4,000	[91]

approximation [73]. Boutorh & Guessoum mentioned three important advantages of using ANNs: high generalisation capacity, the capability to work with incomplete knowledge, the robustness of the generated models and, as mentioned before, the possibility of applying them to complex problems [16].

Khan et al. used ANNs, probably for the first time, to classify four distinct SRBCTs in children. The four distinct types included neuroblastoma, rhabdomyosarcoma, non-Hodgkin lymphoma and the Ewing family of tumours. The authors used gene-expression data from complementary Deoxyribonucleic acid (DNA) (cDNA) microarrays configured for 6,567 genes. Filtering the data left information from only 2,308 genes. For the training process, the authors used 63 samples while the testing used 25 independent examples for performance assessment. The Principal Component Analysis was applied to all the 88 samples for using the ten more important components in the ANNs training process. Then the authors produced 3,750 different linear ANNs by making 1,250 random permutations on the training data. Once trained, the next step used two-thirds of them for training and the remaining for testing. Each third part of the permuted datasets was used as part of the validation or training sets. After training the models, the authors assessed the relevance of the variables by considering the impact that a modification on their expression levels had on the classification error. In this way, they could

detect 96 genes as the most relevant to the four tumour types. These 96 genes were then used in the same process for generating the final 3,750 ANNs models. The authors analysed the principal components for the identification of the ten most relevant. For this, the dataset was permuted 1,250 times and each time two-thirds of it were used for training and one third for validation. The process also included the rotation on the defined partitions allowing the use of the complete permuted dataset for training and validation. These 3,750 models were used for the classification of the remaining 25 samples for testing by voting from all the trained models. The final results showed that each sample was assigned to the correct disease, except for five samples corresponding to completely different types of cancer that were correctly discriminated against. They also find genes specifically expressed in the cancer types, of which 41 had not been previously reported. They demonstrated that the linear ANNs were capable of assessing the correct disease [79].

Posteriorly, Ritchie et al. [145] proposed combining ANNs with Genetic Programming (GP), which they call Genetic Programming Artificial Neural Network (GPNN), for the search of interactions between genetic data and disease. The authors commented that training ANNs using techniques such as hill-climbing tends to get trapped on local minimal. This problem gets worst considering the hypothesis that complex human diseases have a rugged fitness landscape representing the search space [101, 112]. For this reason, they proposed the use of GP for the training process, including the possibility to change the architecture of the ANN. The process includes the use of parallel subpopulations ("demes") in the population. This increases exploratory power and reduces the possibility of ending in poor local solutions. The performance evaluation of the individuals in the GPNN process was through Cross-Validation (CV). The authors compared GPNN against a Back-propagation Neural Network (BPNN). With the BPNN they showed the importance of the network's architecture since the best-classifying BPNN-models had distinct architectures when trained using simulated data from five artificial genetic configurations. The results showed that the GPNN can achieve similar performance in classifying the samples of the simulated Case-Control study, using only two functional SNP[4]. The GPNN also outperformed the BPNN when applied to a dataset with both functional and non-functional SNPs. This was because BPNN tends to overfit the data, showing a better classification performance, but worst prediction performance. Motsinger-Reif et al. later used this approach on three simulated datasets and one real PD data. The results showed that the GPNN can model two and three-locus interactions in moderate sample sizes in simulated data

[4]SNP: A sequence variation on the DNA from a population, is the smallest unit of variation in the genome that must be present in at least 1% of the said population [21]

with tiny heritability values (2–3%). Their model could correctly predict around 60% of the times the disease status in the dataset containing the PD case-control data. The GPNN could also detect an interaction gene-gender, although they mentioned that more research is necessary to address the robustness of the method and to determine how the interaction might influence risk in PD [117].

In a later work, the authors Montinger-Reif et al. tried to improve the GPNN performance by proposing training ANNs with Grammatical Evolution (GE) [114]. The authors called this method Grammatical Evolution Artificial Neural Network (GENN). Having only two nodes in the binary trees is a disadvantage of GPNN, which might not be sufficient in power for more complicated ANNs; also, changes in GPNN require program code updates, affecting the development and flexibility. By using GENN this is no longer necessary, as GENN makes modifications on the ANNs instead of the code. The authors compared GENN against GPNN, BPNN and a random search in simulated data with ten SNPs, two of which were functional. The GENN had similar configurations as GPNN and both had better performance than BPNN and the random search. Furthermore, GENN showed a performance in similar to that from GPNN, but with faster development and more flexibility. In a posterior work, Motsinger-Reif et al. made another comparison between GPNN and GENN [119]. They used simulated data along with HIV immunogenetics data that contained 100 SNPs. The aim of using HIV data was to determine if the techniques could identify interactions showed by an exhaustive search technique. In the simulated data, GPNN required 150 more generations to produce similar performances as GENN. In the HIV dataset, GENN could show a statistically significant two-locus association, while GPNN only identified a single SNP as statistically significant. The authors also mentioned that the GENN computing time increases linearly for the number of variables included, which not holds for the GPNN. However, they also mentioned that more research is necessary to determine power and type I error on data for genome studies.

A posterior study by Motsinger-Reif et al. compared the performance of GENN against Multifactor Dimensionality Reduction (MDR), RF, Focused Interaction Testing Framework and two LR based methods: stepLR and explicit LR (eLR). The authors made a comparison of 3,000 simulated datasets. With eLR, a "positive control" served on assessing the strength of the genetic signal in the artificial data, with the known simulated effect explicitly modelled. The results showed that even eLR had lower performance in the models with two and three locus interactions with lower heritability[5] (1%). This showed how difficult it is modelling

[5]Heritability: the ratio of the genetic component to the total phenotypic variance including the environmental component, the proportion of genetically controlled variation in a population [167].

statistically such interactions. All the techniques had a poor performance on the two and three-locus interactions. All the methods had similar good results when the odd ratio where 1.8 or 2. When the value of heritability was 5%, GENN and MDR had similar performance as eLR in the one and two-locus interactions, but only MDR was better in detecting three-locus interactions. This comes from a more intense search approach of MDR but at a computational cost. Also, GENN and MDR outperformed the Focused Interaction Testing method in models with epistatic interactions. In the case of RF and stepLR, those methods had good performance detecting main effects, but both could not detect purely epistatic models. The authors remark the fact that GENN results in the single and two-locus models were comparable to MDR, but with a computational advantage [118].

Another approach used by Günther et al. comparing ANNs with LR was realised in [58]. The authors generated six different artificial models for gene-gene interaction. The simulated data contained information on two risk scenarios (high and low) and the presence of a disease genotype. The models had two-locus involved, therefore variable selection was not a problem to address. Assumptions for both loci were that the linkage equilibrium [6] and the Hardy-Weinberg equilibrium [7] held. Each model included 100 case-control samples. The authors used six distinct ANN architectures, changing only the number of hidden units, ranging from zero to five with only one hidden layer. For the LR, five models were selected: the null model, three main effect models (only locus A, only locus B, both main effects), and a full model including both main effects and an interaction term. These models also used two variants, one with design variables and the other without them. In the former, the training process fit the LR models to the data with two dichotomous design variables representing each locus. These two variables reflected the heterozygous genotype and the homozygous genotype with two mutated alleles. The measure for evaluating the models was the error arising from the differences between the penetrance matrix generated by the techniques and the theoretically simulated penetrance matrix[8]. In most of the twelve models, the penetrance matrix was estimated most accurately by the neural networks. LR performed better on the multiplicative model and in the dominant epistatic model on the low-risk scenario. But since the underlying disease model is usually

[6]Linkage disequilibrium is the difference between the observed frequency of a particular combination of alleles at two loci (plural of locus showing a specific and fixed position on a chromosome) and the frequency expected for random association [146].

[7]If the genotype frequencies from two alleles (alternative gene configurations that may occur at a given locus) are consistent with being independently sampled from an allele population, then they are in Hardy-Weinberg equilibrium [29].

[8]A penetrance matrix resumes the probabilities of presenting a particular phenotype (e.g. having a disease) considering some specific genotype [28].

unknown beforehand, the ANNs would have a better chance to model the penetrance matrix. The authors also evaluated the ability of MDR and the LR approaches to detect the interaction defined by the four two-locus disease models. The LR mostly selected the one main effect model to represent the multiplicative model. For this reason, LR was not capable of detecting correctly the multiplicative and epistatic interactions, which was also true for the MDR approach.

Considering some drawbacks on the use of ANNs, such as their tendency to over-fitting and the difficulty for obtaining significant information when used as a black box, Beam et al. proposed the use of a Bayesian Neural Network (BNN). The authors based their proposal on the fact that weight decay is a well-known approximation to a fully Bayesian procedure. For this, they used the Hamiltonian Monte Carlo algorithm as a stochastic sampling technique to draw samples from the posterior distribution. This approach together with the use of parallel computing allowed the method to scale on larger databases. For the prior distribution necessary in the Bayesian Network, the authors used a specific prior structure known as the Automatic Relevance Determination. The comparisons were made using 2,000 case-control artificial datasets. The data were simulated with different values for heritability, minor allele frequency[9], base-line risk and effect size. It included 1,000 SNPs for the first test, with two as causal SNPs, and four different effect scenarios: additive, threshold, epistatic and epistatic without main effects. The first comparison was made between BNN, Bayesian Epistasis Association Mapping and the χ^2 test. In the additive and threshold models, BNN showed a consistently better performance than the Bayesian Epistasis approach and the χ^2 test. A second comparison included MDR and Gradient Boosting Machines. The data for this test was simulated with 50 SNPs but without marginal effects. In this comparison, MDR had the best performance, but it was not possible for its use in the GWAS because of the complexity of the method. BNN did well across the tests, surpassing the other methods in most of the genetic models, and it could scale to GWAS-size data. After testing the BNN in simulated data, the authors applied it to a study with information on TB. This dataset contained 16,925 SNPs from 104 infected patients with or with a latent form of TB. In this test, BNN identified five SNPs with highly important interactions for TB. However, an SNP previously reported as relevant ended in the rank 31st with BNN. Because of the small sample size, the authors could not determine on which SNPs would be more relevant in a larger study, considering the selected by BNN or the previously reported in the articles in the work [6].

[9]Minor allele frequency is the ratio at which the allele less represented occurs in a population [21].

2.2.2 Bayesian Probability

Han et al. proposed a method for improving the Bayesian probability network in the search of relations between biologic factors and GWAS in [62]. Their proposal's primary objectives were to address two of the major problems faced in GWAS: the big number of variables and that the studies usually involve a small number of samples. For this, the authors used EpiBN which works as a detector for epistatic interactions[10] based on the use of Bayesian Networks. For better performance, EpiBN used a proposed adaptation measure called EpiScore. EpiScore helps in the learning process by determining a value on how much the network is getting adapted to the data. This could help in the search process of EpiBN driven by a score-and-search approach for model selection. The authors used a screening process for the reduction in the data dimension. For this, they used the Markov chain Monte Carlo approach through the use of the Metropolis–Hastings algorithm. In this way, selecting the relevant variables allows for a reduction in the EpiBN search space. Experimentation took place on four simulated datasets with different interaction degrees among the factors. EpiBN showed a better performance than the Bayesian Epistasis Association Mapping, MDR and SVM. EpiBN also had better performance after changing the sample sizes and the number of SNPs, showing its escalating capability. The authors also compared among EpiScore, Akaike and the Bayesian information criterion in evaluating the adaptation value. The comparison using the proposed Bayesian probability network showed that EpiScore had a better performance. EpiBN also surpassed other feature selection techniques used in previous works. Lastly, the authors applied EpiBN to a GWAS containing data of AMD, LOAD and ADA. EpiBN could detect one highly important two-SNPs interaction on AMD. One of the SNPs was previously associated with AMD while the other one is a candidate genetic factor. In the case of LOAD, EpiBN identifies the APOE gene as important for the disease risk. It also showed a two-SNPs relation as a candidate for an increase in the risk of LOAD. However, EpiBN could not detect a ten-SNPs interaction previously reported. In the ADA dataset, EpiBN achieved the identification of a three-SNPs interaction as a candidate for future studies.

Jiang et al. proposed the MBS-IGain for detecting interactions among genetic factors. Their approach, based on the Multiple Beam Search algorithm (MBS), tried to face the difficulties in the detection of predictive variables when they can present little marginal effects. This is relevant because of the thought that much of the genetic risk for a disease come from

[10]Epistasis is the interaction of two or more genes affecting a phenotype. Statistical epistasis considers the average effect of substitution of alleles at combinations of loci, compared to the average genetic background from the population [29].

epistatic interactions with little marginal effects. MBS-IGain makes the interaction assessment combining a Bayesian score and Information Gain (IG) in the forward selection of SNPs and the definition of the Bayesian network. The MBS-IGain process changes the scoring assessment for the use of IG to select the most informative interactions among variables while using a score criterion to stop the addition of irrelevant SNPs. The authors compared MBS-IGain, MBS and another method called REGAL using 100 simulated datasets. MBS creates models by forward selecting the attributes that improve the score on how well the Bayesian network fits the data. These models are then reduced by removing the attributes that allow improvements on the said score. In both cases, the iteration is repeated until no further improvements can be achieved. On the other hand, REGAL uses a similar approach with the difference that each time it deletes the variables in the highest-scoring interaction. The simulated data included three different interactions: two between 2-SNPs, two among 3-SNPs and five among 5-SNPs from a previous study. The authors also used MBS-IGain on a LOAD dataset which originally had 312,260 SNPs. For this, different subsets were generated considering the 1,000, 5,000 and 10,000 1-SNP models from the APOE gene considering the highest scores. The authors used these SNP sets along with the ones from the 100 SNP models from the remaining SNPs as input in the MBS-IGain. After determining that the best score criterion for the MBS-IGain was the Bayesian Dirichlet equivalent uniform score, the authors compared how well it detected the involved SNPs and the correct interaction. The comparison was against the MBS and REGAL. The results showed that MBS-IGain was more than capable of detecting important SNPs while identifying the correct interactions. A further comparison against seven other methods on the same dataset (LR, MDR, IG, among others) showed that MBS-IGain had a better or similar performance. The authors also tested MBS-IGain on the real LOAD dataset, experimenting with the Bayesian Dirichlet and the Minimum Description Length scores. From the results, the combination with the Minimum Description score identified interactions involving the APOE gene that has previously been reported as relevant for the disease. On the other hand, the use of the Bayesian Dirichlet score allowed the identification of interactions among four and three SNPs. All the most important models included the gene APOE or APOC1. The results also showed that the detected interactions increased the information when considering individually the genes detected as previously more relevant genes (i.e. APOE and APOC1) [76].

2.2.3 Data mining approaches

Ritchie et al. proposed MDR trying to address some drawbacks of the parametric-statistical methods. The limitations include the difficulty in detecting interactions, the need for pre-defined models, the need for specific hypotheses, among others. These limitations are highly relevant when dealing with information from complex diseases which are thought to involve important interactions. The authors based their method on previous work on combinatorial partitioning capable of data reduction. The MDR process involves the search of relevant combinations of variables with appropriate capability for predicting disease. Dimension reduction is also part of the MDR-process by identifying the variables more related to the disease. Once the most relevant had been defined, MDR can suggest low and high-risk groups. For this, the ratio of case-control is assessed considering the number of individuals identified by every variable-value combination. The MDR-process starts selecting a defined number of n variables from all the available ones. Then all the combinations among these are represented in cells showing the variables and their values. The next step involves determining the cells related to high or low risk. For this, the number of identified affected instances are compared with the unaffected ones for each cell. If the ratio between these exceeds a specified threshold t (e.g., $\geq$1.0), then the cell is labelled as "high-risk", otherwise it is identified as "low-risk". The next steps involve selecting the combination of variables with the best classification capability when used on a testing set. MDR defines the training and testing sets by using a 10-CV process. The final accuracy for each model is calculated as the average of the folds. The model (i.e. a specific combination of variables) selected most frequently is then defined as the best model. This allows identifying the number of times that the best model is selected in the whole CV process, indicating a CV-consistency. This consistency reflects the reasoning that a true interacting signal would be detected in most of the folds with disregard on the data partitions, further supporting the importance of the selected variables. The authors used MDR on artificial and on real data. The simulated data included the use of interactions among two, three, four and five SNPs. The detection of said interactions was tested considering up to ten SNPs, including the ones defined as interacting. MDR could correctly detect the interactions in most of the simulated datasets and models. The real data reflected genetic information from ten SNPs from breast cancer patients and controls. MDR could detect a four-way interaction that LR could not identify. Furthermore, LR did not detect even any independent main effects [144].

Rodriguez-Escobedo et al. applied other data mining approach for assessing the association on a real dataset related to haematological malignancies. The information described 17 Killer-cell Immunoglobulin-like Receptors (KIR) genes. Since the association of the KIR-haplotypes is

more relevant than that from individual genes, the authors focused on detecting their interaction with affected cases. The authors used the Fisher's exact test as the first approach for assessing the statistical association. This was done considering both univariate and multivariate statistical tests. From the univariate analysis, the authors only found two variables as associated with the clinical status. However, they are not part of any haplotype. For the multivariate statistical analysis, a first revision on the combinations allowed identifying 336 found as associated considering a $p < 0.05$. These were reduced to 35 when the authors considered as significant those values smaller than $p < 0.0001$. After that, only those combinations associated with a KIR-haplotype were considered for further comparisons, yielding 16 final combinations. The C4.5 algorithm, on the other hand, was used for the induction of a decision tree identifying the variable-value combinations capable of discriminating against the affected class. This tree involved three variables and most of its leaves identified unaffected donors. Similarly than in the univariate statistical analysis, this tree did not involve any of the KIR-genotypes, and for that reason was excluded from further comparisons. Finally, applying the Apriori algorithm allowed the generation of 71,006 rules, of which 12,052 had the affected class as a consequence. Since 24 of those rules were present in a meaningful proportion of the cases, they were selected for further analysis. This allowed identifying one highly relevant rule that could resume the other 23 rules and therefore was considered as the most important. Said rule allowed the identification of affected cases by considering the presence of seven genes and the absence of one of them. The union of these genes, in turn, describes the KIR-haplotype A (cA01|tA01). The authors compared the Apriori-rule and the best combination from the multivariate analysis using a χ^2 test. The comparison suggested that the Apriori-rule was better at determining the affected cases [148].

2.2.4 Hybrid approaches

Hardin et al. used LR as part of a classifiers ensemble which improved the discrimination between Monoclonal Gammopathy of Undetermined Significance and MM. The authors attempted the determining the best classification method for discriminating among Monoclonal Gammopathy, MM and those from healthy donors. On the other hand, they were also interested in detecting the most relevant genes capable of differentiating among the different cases. The principal reason for this is that while the Monoclonal Gammopathy is a relative asymptomatic condition, MM is normally a fatal disease. For this, the authors made a comparison on the capabilities of LR, C5.0, an ensemble of simple decision trees, Naïve Bayes, Nearest Shrunken

Centroid and SVM using real data. The dataset was generated using the information on 12,000 genes and clinical status. Two different approaches were used for selecting subsets of the information, one considering the 10% of them with the lowest p-values. The second was the use of a forward selection function and was initially made up of 500 genes. The initial results showed that even when every algorithm could differentiate between Monoclonal Gammopathy vs. healthy and MM vs. healthy cases, none of the proved clear capability for discriminating Monoclonal Gammopathy vs. MM. Using an ensemble considering the five approaches did not show an important improvement in the discrimination between Monoclonal Gammopathy vs. MM. On the other hand, each approach allowed identifying the different relevant genes. However, there were not coincide with the identified features. What is even more, these were very similar in the Monoclonal Gammopathy and MM cases. Nevertheless, their work allowed identifying relevant genes linked to each case [63].

Tong & Schierz also used an Evolutionary algorithm, Genetic Algorithm (GA), for the training process of ANN and selection of relevant sets of genes. The authors commented on the fact that selecting genetic markers is a complex task. For example, depending on the normalisation process, distinct sets of genes might be selected. This is also the case with different filtering or imputation processes and is further hardener an imbalance in the data. For these reasons, the authors decided on using an approach based on unprocessed gene expression from a DNA chip. They applied the hybrid method to two datasets with information on SRBCT and leukaemia. The authors' proposal used the evolution for two goals: generate models capable of discriminating among the classes and identifying the more informative features. For these reasons, the GA process, on the one hand, evolves the weights in the ANNs models. The evolution also happens for the set of variables involved in the training of the Multi-Layer Perceptron (MLP). The process starts with 300 chromosomes including ten genes each. For the ANN, the authors used MLPs with five units in a hidden layer and 2-4 output neurons. The GA process made use of a tournament strategy for selecting two candidates, a single-point crossover operator and an elitism scheme. The authors repeated this process for 3,000 times for the ANN weights adaptation and the entire process repeated over 5,000 interactions. The resulting sets from the authors' approach were then compared with some previous studies and used with the classifiers MLP, RF and SVM. The classification results showed that the selected sets were useful for identifying the cases. They also suggest that the overlapping genes, those selected by the method and those from the previous studies, might have a more plausible biological significance. On the other hand, they showed that assigning the SNPs' importance based on

classification performance might overlook some important factors since the machine learning methods are designed for a better generalisation [170].

Tong et al. proposed improvements over their previous work considering that information obtained from methods as correlation analysis is not enough by itself for informing on biological significance. For this reason, and to increase the information on the identified factors that might have high biological significance. For this, the authors proposed using an Artificial Neural Network Inference, originally used for modelling interactions between proteins. The idea is that ANN Inference can explain the expression of a biological marker using the other available markers if these can explain a status (e.g. a disease state). In this way, ANN Inference can show the relations among genes in the same pool, identifying how they interact while discovering new interactions by calculating the influence of multiple variables upon a single one. Having this in mind, the authors used an MLP with three layers, repeating the process using each gene as output while the rest were used as input. Once all the models were trained, an average on the weights was calculated, which helped in defining the relation type (i.e. excitatory or inhibitory). The training process for the MLPs models used Monte-Carlo cross-validation repeated 50 times, intending to diminish over-training. The authors iterated the entire process ten times per marker. This approach was applied on a simulated dataset and with SNPs selected in a previous work by Tong & Schierz [170]. The results of the simulated data showed that ANN Inference could detect the most relevant features. It also showed that the use of only two units in the MLP-hidden layer was enough for the training process. Considering the real data, the authors used 96 genes previously selected from the SRBCTs dataset for calculating interactions. After disposing of the least significant relations, the authors generated a genetic map using the program Cytoscape. The map included only the most significant relations of each gen. These relations were proposed as possible markers for future studies. This map allowed the authors to make some predictions on the presence of the cancer type, considering the interaction among the markers [169].

Boutorh et al. proposed another approach trying to identify complex interactions among SNPs on data related to ASD, BrC, CC and MR. For this, the authors combined a pattern recognition approach involving feature extraction as an important factor. The hybrid technique combines the approaches GA-ANNs-GE-Association Rule Mining (called by them as GA-NN-GEARM). This approach combines mining association rules with GE for the feature selection, making a faster and less expensive process. This also allows identifying the SNPs related to the cases and those relevant for the healthy instances. The authors then used the union of the selected SNPs as input for training ANNs models while helping to address the dimensionality

problem. On the other hand, the use of ANN allows for better classification performance. They used and compared three types of ANNs: a two-layer feedforward network with a sigmoid transfer function, a Radial Basis Network with the Gaussian activation function, and a Focused Time Delay Neural Network. Finally, GA was used to evolve the parameters of the different techniques, such as the number of generated rules, neurons for the ANN hidden layer, the maximum number of iterations for the ANNs and the number of times the ANNs will repeat the training process with different initial weight values. In this way, GA makes a metaparameter optimisation allowing for a better operation. The performance of the GA-NN-GEARM was similar or better than other techniques combinations previously used. The combined approaches included feature selection using Distance Discriminant, ReliefF, based on R-value and Feature Clearness. For classification, the authors used the following approaches: k-Nearest Neighbours, SVM and the Artificial Gene Making method. The GA-NN-GEARM performances were better most of the time than those from the combined approaches, or at least as good. The number of selected SNPs was also smaller in most of the GA-NN-GEARM results. With the GA-NN-GEARM learning approaches, the best results were obtained using the Focused Time Delay approach, most of the time with a smaller number of features. The second-best performance was achieved with MLP, although generally using more features. The results do not show if there was an over-training when using the Focused Time Delay method, in which case the MLP would be a better option [16].

2.2.5 Logistic Regression

To find a polygenic model that could help to identify SCZ cases and the detection of the relevant SNPs involved. The authors of the study gathered data considering some SNPs previously associated with patients with an autistic spectrum disorder, bipolar disorder and SCZ. The polygenic model describes an additive model where the sum of individual signals determines an expressed phenotypic trait (e.g. a physical condition). After a quality procedure, two out of 84 SNPs were dropped from the dataset. In the same manner, some samples were removed altogether after revising the data quality or ancestry. For this reason, the final dataset had 2,783 controls and 2,964 patients. The authors made the analysis in two parts: i) assessment on the Hardy–Weinberg equilibrium and the possible association SNPs-SCZ; ii) the use of resampling for generating training and testing sets for calculating polygenetic scores. For the first part, the authors used two different methods for the statistical tests: LR for the first part and a log-additive model for the second. LR under five different genetic models was

employed for association assessment. The results from the first part showed that most of the SNPs' risk tendency coincides with some of the previously reported, considering p-values under < 0.001, and with all of them for p-values below <0.05. Furthermore, at least an SNP was found associated with SCZ using the trend test, allelic and codominant genetic models, after correction for multiple testing. The log-additive model of the second part involved the use of LR for generating log odd ratios which later were used on the training test. This process helped in assessing if an increase in SNPs numbers included in the models could help for better explaining heredity. The authors repeated this process 1,000 times, always partitioning the data in halves, one for training and the other for testing. Each SNPs included in each polygenic model was selected considering a p-value threshold. As part of the process, this threshold value changed for the increment on the SNPs included in the models. The values were 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1. The authors also used three different ranges for the experimentation 0.05–0.2, 0.05–0.5, and 0.2–0.5. Using said thresholds delimited the number of SNPs to include in the models. Once the authors generated the models, they assessed the association between scores and case status considering their p-values distribution when applied on the test sets. Similarly, the classification performance was measured using the Receiver Operating Characteristic generated using said models. The results from this second stage suggested relevant SNPs and their influence on the Area Under the Curve measure for classification. However, these scores did not achieve significant p-values for their association with SCZ. In the classification's case, the results showed that the increase in the number of factors included for the score had a positive effect on the classification performance. Although said perforce was better than that coming from a random process, the authors did not find it that much better [72].

Li et al. combined machine learning and regression-based methods for the search of interactions among SNPs on data from case-parent trios related to CL/P. The authors' goal was identifying gene-gene interactions using previous GWAS and by considering candidate loci to make a more treatable problem. The idea is that detecting the cumulative modest effects of interacting genes might explain the "missing heritability"[11]. Nevertheless, even the limitation on the regions of interest does not allow for an exhaustive two or three-way interaction search because of the resulting highly stringent significant threshold value after corrections. The authors used data on 346 SNPs from 895 trios of Asian ancestry, 395 SNPs from 681 trios of European ancestry. They also used the data to generate pseudo-controls

[11] "Missing heritability" is the difficulty to predict the effects of identified genetic factors from GWAS studies since these might include a high number of independent factors with small main effects.

for the training process in the machine learning methods and for imputing missing data. By employing three different approaches the study evaluated the interaction of the genes by using machine learning extended for the case-parent trio design and statistical tests applied only to the cases. Considering advantages such as the capability for detecting non-linear interactions, the needless of pre-specified models among others, the authors selected RF as one of the machine learning approaches. Additionally, the boosting process in the RF helped in controlling the false-positives, while the frequency presence of specific features in the RF's trees suggested on their importance. With this approach, the RF method delivers a list suggesting variable importance by their impact on the classification performance, which helps for generating a relevance ranking. The authors made a comparison against the classification on the same examples before and after permuting the values on each variable from the list. For the RF process included generating 1,000 trees containing at most the squared root value of the number of variables in them. The second machine learning approach combined LR augmented by simulated annealing for the search of the best combinations (logic term) considering the goodness-of-fit measure from the regression model. The logic term describes combinations of variables considering connectors such as AND and OR, as in $SNP_1 \wedge SNP_2$. In this way, only two different cases needed assessment: *true* if the condition is present and false otherwise. The LR step involves learning the covariates using the specific logic terms. Acceptance of new logic terms is based on the use of simulated annealing. For this, a comparison is made between the LR model and a conditional regression model which is more appropriate for matched data. If a new term improves reducing the difference between them, then it is accepted. Otherwise, the new logic term acceptance depends on a probability. The authors applied this process considering all the SNPs with a limitation of three or five predictors for the logic term. They applied the statistical tests considering only cases with the idea that if two SNPs appear as correlated it would be easier to detect their interaction by a parametric approach. The tests evaluated the correlation among SNPs and then used a χ^2 statistic with one degree of freedom for determining statistical significance. The independent results from each approach were then tried using conditional LR for testing the two-way gene-gene interactions. These were measured by departure from the predicted effects considering a log-linear model. The results from the RF allowed identifying 25 and 32 SNPs for the Asian and European ancestry, respectively. The results from LR allowed identifying two and three-way interactions. The results from the statistical tests allowed the identification between two pairs of SNPs. These results came from running the approaches using the data corresponding to each ancestry. The methodology allowed the identification between different pairs of genes, having SNPs that showed only marginal effects. Similarly, three regions

with only marginal effects on the CL/P also showed interactions. These interactions were, in general, different in each ancestry [92].

2.2.6 Support Vector Machines

Cuevas-Tello et al. used SVM for generating models capable of discriminating disease status. They also used Iterative Dichotomiser 3 (ID3) for suggesting the relevant variables. A first step involved evaluating the SVM for discriminating between affected (i.e. patients) and unaffected cases (i.e. controls). This involved the use of a GA in the search for two of the SVM's parameter values. In this way, the authors searched through different combinations for the soft margin (C) and Gaussian functions' width (σ). The best parameter values found were determined after the analysis of their classification performance. The search involved the use of two artificial datasets and information from one real dataset. The real dataset included information from haematological diseases on 23 variables, representing genes (16 variables), one combination (1 variable) and haplotypes presence (6 variables) from 43 patients and 300 controls. The first artificial dataset included only random generated data on 43 and 300 affected and unaffected cases, respectively. The second artificial dataset combined the values from the real dataset variables while assigning the class randomly. Using SVM optimised by GA allowed a mean accuracy performance of 93.5% on the real data after 18 different data partitions were used. Using SVM on the artificial data helped in determining the appropriateness of the method for the case classification using genetic variables interactions. On the other hand, the authors used ID3 for the generation of a graphical representation of the model since SVMs this is not a simple process. For this, the authors generated three different trees considering 30%, 50% and 75% of the data for training. These trees allowed suggesting protective effects previously published and mainly related to KIR genes or haplotypes rich in KIR genes having Human Leucocyte Antigen (HLA)-C specificity [32].

2.2.7 Tree-Based approaches

Using a tree-based approach, Li et al. proposed a permuted-RF method. This used the RF combined with a permutation process. The idea is to find the combinations between genes that allow explaining the expressed phenotype without relying only on the main effects. Traditional statistical methods generally focused on these main effects which reduce their capability for detecting interactions. The approach is based on assessing the effects on the identification of the classes after permuting the information on two SNPs. If there is an interaction between the

SNPs, its removal will greatly affect the classification. Since the detection of interactions is not explicitly part of the RF process, the authors identified these by using RF Networks. The process starts by training an RF model with all the features. After this, a set of tests is applied per each feature pair. These pairs of features were used for generating permuted data per class. Two different approaches were applied: i) permuting the two selected features independently; ii) the permuting with both features together. Then the permuted datasets were used for testing on the trained model for the assessment on the classification errors. The difference in the errors allowed to identify important interactions between the features reflected because the error will increase if the features are permuted independently. The authors compared the permuted RF method against other techniques, such as MDR and Statistical Epistasis Network. For this, they applied the methods to two datasets containing genetic sequences of BlC patients. One of the datasets contained 7 SNPs and the other 39 SNPs. In the dataset with 7 SNPs, the permuted RF could identify the most important pair of interaction detected by MDR. In the dataset with 39 SNPs, the comparison was between a previously constructed network using Statistical Epistasis. This described the interaction between the polymorphisms. For the comparison, the authors partitioned the permuted RF results into three groups. The groups in the maps included the same interactions generated by the permuted RF and those present in the Statistical Epistasis Network model with levels of 28.57%, 84.62% and 70.0% for each group. The permuted RF also identified new strong interactions for future research [91].

The proposal by Turk et al. was made since the definition of HIV progression based on biomarkers is still an open matter. For this reason, the authors proposed the use of decision trees for identifying possible relevant biomarkers. The motivations for the use of decision trees are that they can work with small datasets and deliver explainable models. The work searched for variables that could help in identifying patients with different HIV-disease progressions. Some of the most used indicators on the HIV-infection progress are the baseline Cluster of Differentiation 4 (CD4)+ T cell count and viral load. However, other indicators might have been overlooked and therefore were included in the study. The authors applied their proposal on data from 75 patients and 21 healthy donors and included 88 variables. The source of these was: 16 clinical, 10 immunological, 10 genetic, and 52 related to soluble plasma factors. They used three different subsets sizes for the experiments: i) smallest subset: 27 instances including information on the plasma soluble factors (52 variables); ii) medium subset: 48 instances expanded the previous smallest one by including the data from genetic and immune responses (20 more variables); iii) the complete dataset with 75 total instances and the clinical information. Additionally, the authors used three different class definitions: *C1* considering

the 350 CD4+ T cells/μL during the first year as a threshold; the other two classes considered the viral load value of 100,000 copies/mL as a threshold at baseline for the class *C2* and after six months for defining the *C3* class. The process included assessing the individuals or constructed variables and their association with the defined classes. For this, the authors used different statistical tests, a feature selection method and J48, a specific implementation of the C4.5 algorithm. Using the Mann–Whitney test, six variables representing chemokines and cytokines (from the smallest subset) were associated with the initial baseline viral load. This was the case when the authors compared the data from patients against that from healthy donors. Similarly, using Spearman's test for correlation, four variables were found related to the baseline immune activation (percentages of activated CD4+ T cells). Four of the six variables detected as associated also correlated with the baseline viral load. The authors made a second analysis was made for assessing the association with disease progression using correlation-based feature selection. The results showed correlations of two variables with the class *C1* on the three subsets, two with *C2* on the smallest subset and none with *C3*. Baseline CD4+ T was correlated with both classes *C1* and *C2*. Another process using the combination of correlation assessment and the J48 algorithm was applied to identify the variables that impact on disease progression. The authors induced decision threes with J48 using the three different subsets and the class definitions *C1* and *C3*. The clinical information, more specifically the baseline CD4+ T cell count, dominated all the other variables, being selected as the first test for the class of *C1*. Another variable manually defined was the only test for the class *C3*, this considering the removal of the clinical variables. The result showed that the count of CD4+ T cells at the first enrolment was the best indicator to identify the patients that eventually fall below 350 cells/ml [173].

2.2.8 Some problems in previous studies

The search for the causes of complex diseases using genetic data has been facing several problems. Some of these are presented next by considering some approaches used.

Some problems of using ANNs include the need for selecting the right parameters for training, a process that would require a great number of experiments. Another way to search for these parameter values or to make the training process is by using metaheuristics algorithms such as GA and GE. However, this increases the computational burn and the complexity in the search due to additional parameter values [42, 95]. Moreover, the training process could also include changing the architecture of the network, thus increasing the search space. Furthermore, one of

the most used training methods based in the gradient descend, the traditional backpropagation, is prone to get trapped in local minima [8].

Another problem with the use of ANNs is that most of the time, they are used as a black box. In classification, for example, the exact process for the determining the decision in the trained models is not easy to explain due to the high number of interactions in the ANNs (i.e. weights, activations functions and layers) [87]. This situation does not allow a straightforward identification of the main features that have a greater level of influence on the outcome. To overcome this problem, the ANNs have been combined with other approaches, such as data mining or decision trees to make a feature selection possible.

Some major problems with Bayesian probability are the difficulty to select the right prior and the high computational cost needed in the training process. For example, the process for learning the Bayesian Networks architecture requires exploring different variable orders in the network, which implies different effects depending on the specified influences [62, 76].

Logistic regression has advantages such as its low computational cost and capability to detect main effects[12] in genetic data related to the disease. However, the ability to capture main effects reduces their capacity to characterise purely interactive effects, which are believed to be more common in the case of complex diseases [116].

The major advantage of the hybrid approach is that it may help to overcome some shortcomings of applying each method by itself. Nevertheless, such combination and advantages increase the computational costs [116, 16, 91]. However, this approach allows for a better search process and continues to be used, as in this work.

In the case of the decision trees approach, it can be used for feature selection with, depending on the data, low risk of over-fitting. However, some algorithms require the target attribute (i.e. the class to be learned) to have only discrete values. Additionally, they tend to perform well with a few highly relevant attributes, but worse if many complex interactions are present. The decision trees can also have an over–sensitivity to the training set (e.g. on imbalanced data gives preference to the more represented class), to irrelevant attributes and noise [149].

Some major disadvantages of MDR are that it can be computationally intensive, its output models difficult for interpretation and the prediction can be hampered when the dimensionality of the best model is relatively high and the sample is relatively small [144].

[12]The term *main effects* refers that a learned model on an exploratory variable is defined by the individual relevance of some explaining variables. In this sense, the information detected on a few variables is enough for explaining the phenomena. On the other hand, when the interactions are more relevant (*interactive effects*), a more complex model capturing is required.

By using a data mining approach, it is possible to identify important features that influence the outcome of classification. However, this process can reclaim great levels of computational resources, as exemplified by the rules that would be needed with 66 features using the Apriori algorithm with a complexity of $O(2^n)$, where n is the number of features, could generate 7.38×10^{19} possible rules, indicating an exponential growth [166].

Finally, SVM works in high dimensionality data, does not tends to get trapped on local minima, and is robust to noise present in the data [84]. Nevertheless, it cannot be directly used for feature selection. Moreover, support vector optimisation can only be solved analytically or when a very small number of training data. Thus, there is a need to select the best parameters for the SVM by tuning them. This would involve making the selection on: i) a kernel function; ii) a kernel parameter or parameters; iii) regularisation parameters values (C, v, ε). Another limitation is the low training speed when very large data is used [159].

Considering these reasons, for the task of finding associations, where the knowledge on the relevant features is fundamental, it is visible that approaches such as ANNs or SVMs would require additional processes. Although LR is capable of detecting the principal features determining some target output, its process is limited due to the incapability to detect interactions among them, which is relevant when working with genetic data. Another more complete approach is the use of Bayesian probability methods since these have shown being capable of detecting interactions among the features. However, the process for the determination of the network is difficult. Finally, since the determination on the more relevant features is key in the definition of interactions, the use of methods that deliver interpretable models is highly desirable. In this sense, it is well suited to use methods such as the Apriori algorithm, decision trees and MDR. This can be further improved by using methods highly suited for classifying tasks such as ANNs or SVMs. These can help for indirectly testing the classification information existing in specific features and also can help in determining hypothetical classification limits on a specific dataset. In this way, a hybrid approach would take the advantages of state-of-the-art classifiers along with the use of interpretable models in the search of interactions and, finally, in the definition of specific hypotheses for association tests on genetic data.

Chapter 3

Methodology for the suggestion of relevant mutations on Human Immunodeficiency Virus genetic data using machine learning

After reviewing some of the state-of-the-art approaches in the search of associations among genetic variables in Chapter 2, this chapter presents the methodology for discovering combinations of variables related to the Human Immunodeficiency Virus (HIV) data, described in Chapter 1. The methodology involves using the approaches Association Rules Mining (Apriori algorithm), Decision Tree induction (Iterative Dichotomiser 3 (ID3)/C4.5), Data Mining (Multifactor Dimensionality Reduction (MDR)) and Artificial Neural Networks (ANNs) (Multi-Layer Perceptron (MLP)). The first section makes an explanation of the involved algorithms, followed by a toy example on their use. Since this section delves on the basics of the mentioned algorithms, readers who already know these models can skip ahead to section 3.2 where the methodology is presented. The process generates association rules, decision trees and model from the algorithms. The variables involved in them are then further evaluated by considering improvements in the MLP classification performance. The final variable relevance is calculated considering their inclusion on the results from the algorithms and their capability for discriminating between classes. Once the results are analysed, our methodology suggests the best combinations of variables for hypotheses testing and probable biological assessment.

3.1 Description of the approaches in the methodology

3.1.1 Artificial Neural Networks

The human brain, the inspiration for the ANNs, is the most complex, non-linear and parallel computer known to the man as described by Haykin [65]. Ramon y Cajal set the first bases for understanding the brain functions. His pioneering work was the first to refer to neurons as the processing elements of the brain, and also established that the information is transferred from the dendrites toward the axon, travelling through the soma or cell body [140].

The human brain is constituted by approximately one thousand millions neurons with over 104 connections for each of them. Considering a simplification, the neurons have three major components: dendrites, soma and axons. The dendrites (named by their resemblance to the shape of a tree) are the receptive nerve networks that transport the electrical signals towards the soma. In the soma takes place a signal accumulation and the minimum threshold needed for the neuron activation is defined. The axon is a large fibre that transports the signals from the cell body to other neurons. The point of contact between the axon of one cell and the dendrite from another is the synapse, being the more common the one without physical contact between the neurons. It is by the neuron's arrangement and the force of their connections that the neural network functions are established. Figure 3.1 shows a representation of a simplified network with two biological neurons.

As mentioned before, chemical synapses are the most common. These synapses receive a transmitting substance from a presynaptic neuron which is distributed among the entire synapse connection between the neurons. After receiving the signals, they act in generating a new postsynaptic signal. The synapses make the conversion from a presynaptic electrical signal into a chemical signal and later into a postsynaptic electrical signal [158].

The ANNs have shown their high capabilities in a great number of everyday problems. Their proven advantages made them one highly used tool for pattern recognition, function approximation, prediction/forecast, optimisation, etc. ANNs can model non-linear complex systems, and even though other techniques can also do so, ANNs have shown higher flexibility [73]. Haykin defines the ANNs as an adaptive machine with a highly parallel computing process made up by simple processing units [65]. ANNs are a mathematical model that tries to simulate the process made by the human brain and the central nervous system. One of the major goals of an ANN is learning from the environment while trying to maintain a model consistent enough for the desired tasks that are the main interest for its application.

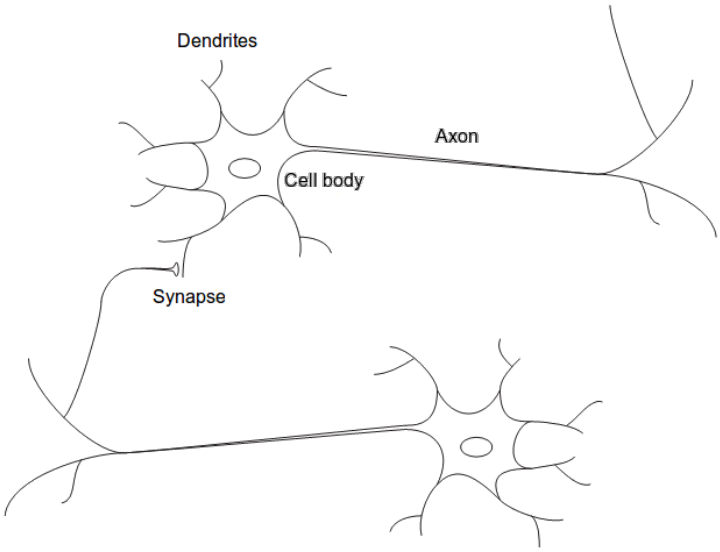


Figure 3.1 Simplified network representation with two neurons (adapted from [59])

Constructing an ANN starts with selecting the neuron model and a specific architecture. The learning process happens by adapting the network's free parameters, guided by the interaction with the environment. Most of the time, the learning process consists in changing the synaptic weights, but it could also include changing the network's architecture. The learning process stores the knowledge in the weights and the ANN keeps it available for future use [65].

Some wanted characteristics present in the ANNs models are:

- Massive parallel computation
- Learning ability
- Adaptability
- Processing of inherent context information
- Error tolerance
- Mapping from inputs to outputs

3.1.2 Multi-Layer Perceptron

The MLP is one ANN approach used for supervised learning on both classification and regression tasks. The MLP is an improvement over the original Perceptron model proposed by Rosenblatt [150]. Generally, the training process for the MLP is done using the backpropagation method [152] based on the gradient descend, although other methods can be used.

The neuron is the basic processing unit in the MLP and its functioning is based in its biological counterpart. The MLP neuron, shown in Figure 3.2, takes a set of inputs that are modulated by a set of weights. These serve as the excitatory or inhibitory signals for the neuron. An accumulation takes place in the neuron that is later combined with a bias that helps in establishing a threshold for the neuron activation. Finally, this combination is transformed by an activation function, usually a sigmoid, which deliver the output value of the neuron. The basic neuron includes three basic parts:

1. A set of synapsis connecting input values to the neuron.
2. A summation point where the weighted inputs values are accumulated.
3. An activation function that limits the neuron's output. Typically, the output value is normalised in the range [0,1] or [-1,1].

The calculation process in the neuron k involves the accumulation point and the use of an activation function. The accumulation is done using 3.1:

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{3.1}$$

while the activation function is done according to 3.2:

$$y_k = \varphi(u_k + b_k) \tag{3.2}$$

where $x_1, x_2, \cdots, x_m$ represent the input values, $w_{k1}, w_{k2}, \cdots, w_{km}$ are their synaptic weights. The accumulation value, u_k, is the lineal combination from multiplying the inputs by their weights. The value of v_k represents the sum of u_k to the neuron's bias or threshold, b_k. Finally, the output value for y is given by applying the activation function φ to v_k.

In the MLP the bias value is represented by an additional, weight $w_{k0} = b_k$ with a fixed input value $x_0 = +1$, changing the simple neuron as shown in Figure 3.3.

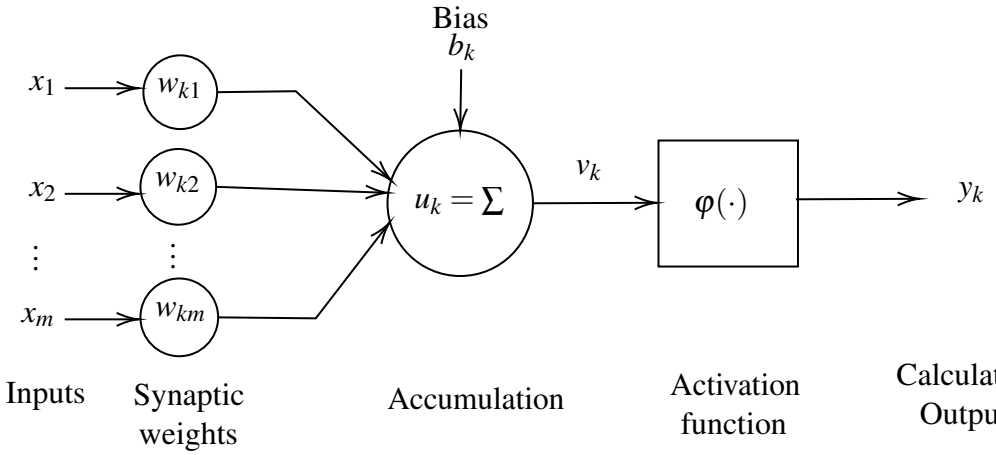


Figure 3.2 Example of a simple neuron having the weights $w_{k1}, w_{k2}, \ldots, w_{km}$ connecting the input values given by $x_1, x_2, \ldots, x_m$ to the neuron k. The point of accumulation, u_k, sums the multiplication of each input by its weight according to Equation 3.1. This sums plus the bias value, b_k, gives the value v_k which is then used as input in the activation function. The bias influences the neuron activation by increasing or decreasing the accumulated value. The value delivered from the activation function determines the calculated output value of the neuron (y_k) determined by Equation 3.2 (adapted from [65]).

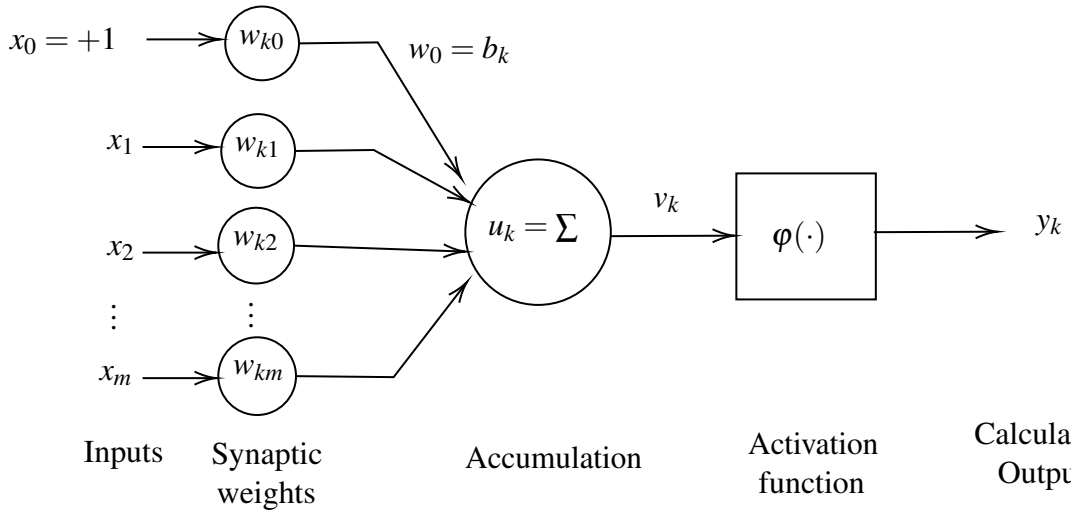
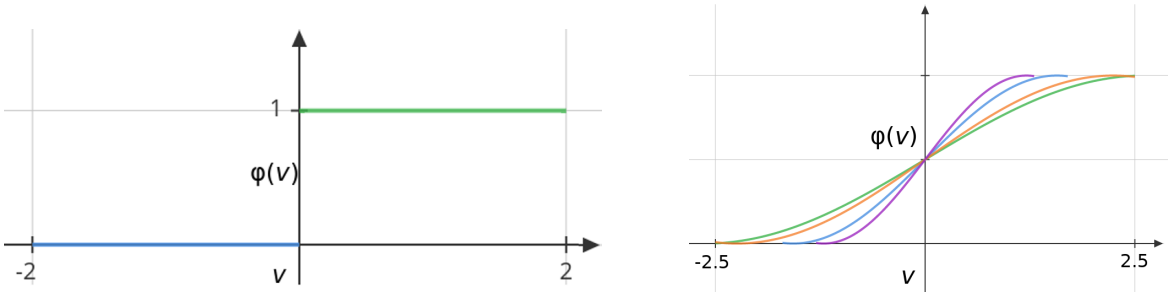


Figure 3.3 Example of a simple neuron having the weights $w_{k0}, w_{k1}, \ldots, w_{km}$ connecting the input values given by $x_0, x_1, \ldots, x_m$ to the neuron k. These values include the fixed value of $x_0 = +1$ and the bias as weight $w_{k0} = b_k$. The point of accumulation, u_k, sums the multiplication of each input by its weight, this gives the value v_k. The v_k is transformed by the activation function and determines the calculated output value of the neuron, y_k (adapted from [65]).

The simplest activation function $\varphi(v)$ is the step function, shown in Figure 3.4a. This function is called *all or nothing* since it outputs a value of 1 if the calculated value is greater than 0, and 0 otherwise. This is defined by Equation 3.3:



(a) (b)

Figure 3.4 Examples of activations functions for ANNs: a) the step function described by Equation 3.3; b) the sigmoid function that can be defined by Equation 3.4.

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 1 \\ 0 & \text{if } v < 1 \end{cases} \tag{3.3}$$

A more complex and powerful activation function is the sigmoid shown in Figure Equation 3.4. This is one of the most used functions in ANNs with a balanced behaviour between linear and nonlinear. One example of these "S" shaped functions is the logistic function defined by Equation 3.4:

$$\varphi(v) = \frac{1}{1+\exp(-av)} \tag{3.4}$$

where a is the parameter defining the curve in the sigmoid function.

The organisation of the neurons in the MLP is given by an architecture, which defines how many neurons and layers are used. The *input layer* is composed by the neurons that take the values for the learning process. The number of neurons in this layer is defined by the dimension of the data. After the input layer, the MLP contains one or more additional layers, which are called *hidden layers*. Each neuron in the first hidden layer makes the accumulation and application of the activation function on the values from the input layer modulated by their weight values along with the bias value. This process is repeated on the neurons of each posterior hidden layer using as input the values outputted from the previous layer's neurons. The *output layer* is the last in the MLP and its neurons deliver the final output values resulting

from the MLP. For this reason, the MLP is a multilayer *feedforward* ANN. Feedforward refers to the fact that the accumulation process and transformations using the activation function takes place on the outputted values from the neurons in the previous layer until the final calculations at the output layer.

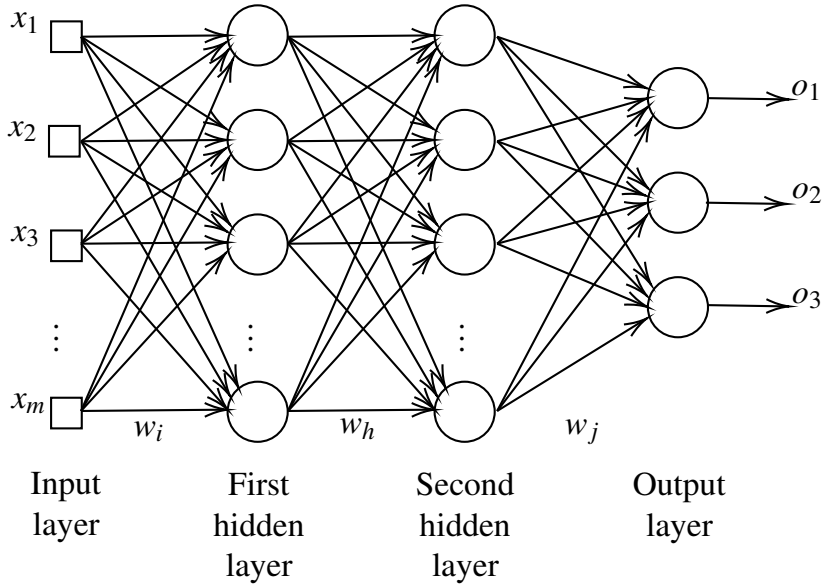


Figure 3.5 Example of MLP with two hidden layers. The input layer is represented by the input data while the neurons in the output layer deliver the values resulting from the feedforward process (adapted from [65]).

As mentioned before, the learning process in an MLP generally is done by adapting the values at the weights, commonly done by using the backpropagation algorithm. In this sense, the data for learning includes the variable values $(x_1, x_2, \cdots, x_m)$ and known class values or targets $(t_1, t_2, \cdots, t_m)$ for each example in the dataset. The learning process is based on adapting the weights according to some correction values. These values are calculated considering the errors at the output neurons with respect to the target values. In this way, the learning process is guided by the target values compared against the outputted values from the feedforward process $(o_1, o_2, \cdots, o_m)$.

The learning process generally is done using the backpropagation algorithm. This process makes uses of activations functions such as the sigmoid in the feedforward phase and the derivates of said functions are used in the backpropagation phase. In this phase, the detected errors of comparing the desired targets against the networks output values are then used in

the modification of the weights. One way for calculating these errors is by using the squared difference between the target and the network outputs by Equation 3.5:

$$\varepsilon_k(n) = (t_k(n) - o_k(n))^2 \tag{3.5}$$

where $\varepsilon_k(n)$ represents the squared difference between the $t_k(n)$ target at the $o_k(n)$ output neuron for the training example n. Depending on the learning strategy, the process might change the network weights after each example used for training, as in online learning, or after showing all of them, as in batch learning.

Updating the weight in an output neuron k considering the neurons in the hidden layer j by Equation 3.6:

$$W_{jk} = W_{jk} + \alpha\varphi_j(n)\delta_k(n) \tag{3.6}$$

where W_{jk} is the current weight value, α is the learning rate, $\varphi_j(n)$ is the activation function used in the output layer. The value for $\delta_k(n)$ is calculated using Equation 3.7:

$$\delta_k(n) = \varphi'_k(n)\varepsilon_k(n) \tag{3.7}$$

where $\varphi'_k(n)$ corresponds to the activation function derivate for the neuron k. In this way the local gradient descend is calculated indicating the direction and steps for reducing the errors.

Similarly, the weights for the hidden layers are changed by distributing the influence on the outputted errors to each hidden neuron by using Equation 3.8:

$$W_{ij} = W_{ij} + \alpha\varphi_j(n)\delta_j(n) \tag{3.8}$$

where W_{ij} is the current i^{th} weight value in the j hidden layer, α is the learning rate, $\varphi_j(n)$ is the activation function used in the hidden layer. The value for $\delta_j(n)$ is calculated using Equation 3.9:

$$\delta_j(n) = \varphi'_j(n)\sum^{k}\delta_k(n)W_{jk}(n) \tag{3.9}$$

where $\varphi'_j(n)$ corresponds to the activation function derivate for the hidden neuron j. This is multiplied by the calculated local gradient descend modulated by the updated weights connecting the next layer. The MLP process can be further improved by adapting the learning rate by a *momentum* term, which reduces the probability of ending in local minima while increasing weight-search stability.

Toy example using MLP learning

Next is presented a toy example of the application of the MLP algorithm. This example uses three different logical tables, representing values for the functions AND, OR, and XOR (see Table 3.1). Two variables, g_1 and g_2, represent the inputs. C represents the target class from the combinations. For example, the combination $g_1 = 0$ and $g_2 = 1$ for the AND table gives the class value $C = 0$. The example includes 20 different values combinations for g_1 and g_2 as input data for each function. In this data, it is visible the relevance of having the correct input values since incorrect or noise values would hinder the learning process. The resulting models for each logical function are shown in Figure 3.6. These results show how difficult it is to easily identify the influence of the variables on discriminate between the classes.

Table 3.1 Twenty examples used as input data for the toy example for each function. The variables are coded as g_1 and g_2. The result for each combination is according to the specified function, and it describes the target Class (**C**) determined by the logical functions AND, OR and XOR. The examples are repeated since there are only four possible ways to combine two binary variables (i.e. 00, 01, 10, 11).

Example	g_1	g_2	C (AND)	C (OR)	C (XOR)	Example	g_1	g_2	C (AND)	C (OR)	C (XOR)
1	1	1	1	1	0	11	1	1	1	1	0
2	0	0	0	0	0	12	0	0	0	0	0
3	0	1	0	1	1	13	1	0	0	1	1
4	1	1	1	1	0	14	0	1	0	1	1
5	0	0	0	0	0	15	0	1	0	1	1
6	1	0	0	1	1	16	1	1	1	1	0
7	0	1	0	1	1	17	1	1	1	1	0
8	1	1	1	1	0	18	0	0	0	0	0
9	0	0	0	0	0	19	0	1	0	1	1
10	0	1	0	1	1	20	0	0	0	0	0

3.1.3 Association Rules Mining

First introduced by Agrawal et al. for determining the relationships among a set of items in a database [1]. Association rules are a form of unsupervised learning that has been applied to many areas such as retail business, web mining and text mining [155].

The most complicated part of the association rules inference is to find the sets that can satisfy the minimum desired requirements, which would determine the goodness of the analysis. These sets are the base for rule inference. As mentioned by Santhiveeran, the source for association

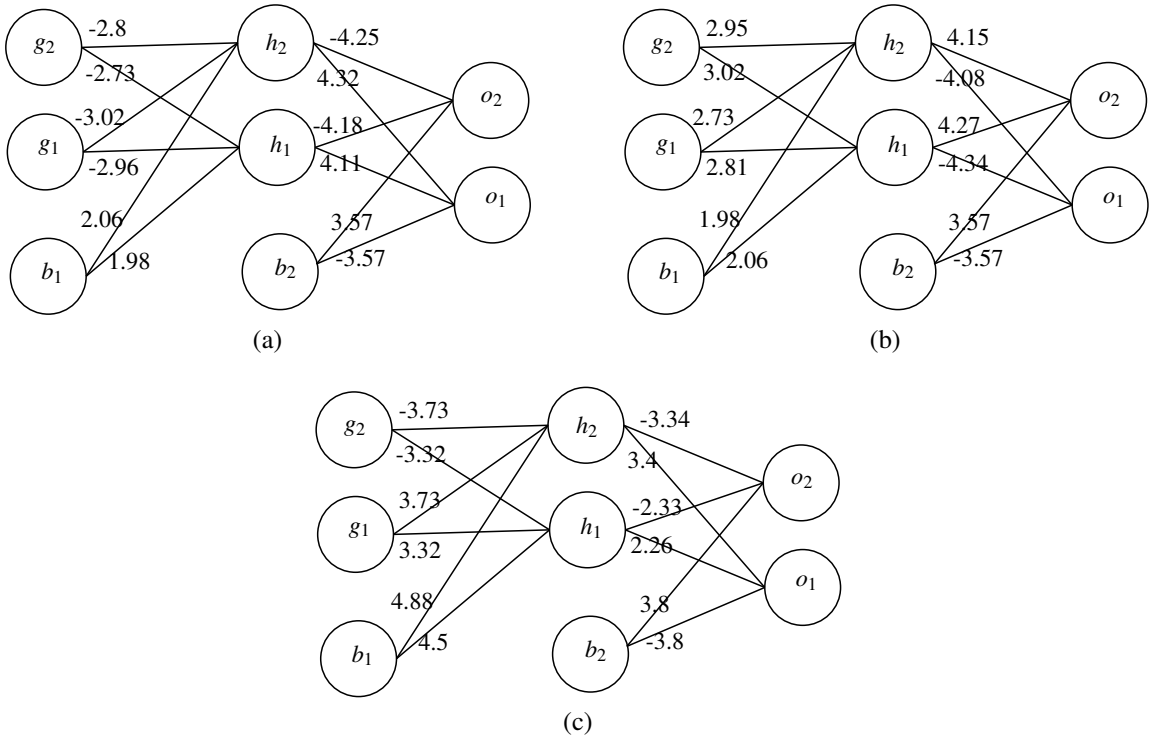


Figure 3.6 Two-variable models after the application of MLP on the input data of the toy example (see Table 3.1): a) shows an MLP for the AND; b) shows an MLP for the OR; c) shows an MLP for the XOR. All the models include one hidden layer with two neurons and two outputs layers, all of them using a sigmoid activation function. The figure includes the learned weights, from which it is difficult to assess the relevance of the variables.

rule algorithms is often a database viewed as a set of tuples that include some combination of different items [155].

An association rule implies that one or more items (a subset of items) are related to one or more items (a different subset of items). Formally it is described as: given a set with m items $I = I_1, I_2, \ldots, I_m$ and a database with n transactions $D = t_1, t_2, \ldots, t_n$, where a transaction contains k items $t_i = I_{i1}, I_{i2}, \ldots, I_{ik}$ with $I_{ij} \in I$, an association rule is an implication $X \implies Y$, where $X, Y \subset I$ are sets of items called itemsets and $X \cap Y = \emptyset$ [77].

Two main measurements help in assessing the robustness of the inferred rules from the data. The first measure is the support (s) of an item or set of items that correspond to the percentage of transactions in which the item can be found. In other words, the support of an association rule $X \implies Y$ is the percentage of transactions in the database that contain $X \cup Y$ [39]. The support is important to reduce the probability that a rule may have occurred by chance [166].

The second measurement is confidence (α), which describes the strength of the co-occurrence between two items or sets of items. The confidence for an association rule $X \implies Y$ is the ratio of the number of transactions that contain $X \cup Y$ to the number of transactions that contain X [39]. The confidence measures the reliability or strength present in the inference made by a rule [166].

Both measurements help in processing the association rules. In general, the algorithms for the generation of association rules have two steps: i) discover the large itemsets (those that have a support value above a desired threshold); ii) association rules are generated by using the discovered large itemsets, including only the itemsets that surpass the desired confidence threshold [155].

Apriori algorithm

The Apriori algorithm is one of the most well-known algorithms for generating association rules. This algorithm was the first that made use of a support-based pruning for controlling the exponential growth of candidate sets [166].

This algorithm uses the property described by the Apriori principle: if an itemset is frequent, then all of its subsets must be also frequent [166]. This indicates that if an itemset satisfies the support threshold, all of its subsets will satisfy it as well. This property helps to discard small itemsets since it implies that all of its subsets are also small [155].

The Apriori algorithm works by generating candidate itemsets of a particular size, which are then further reduced using the support threshold. Only the itemsets considered as large are used for the next iteration. Candidates for the next pass (C_{i+1}) result from joining large itemsets from the current pass (L_i) to the set of large itemsets from the previous pass (L_{i-i}) [155].

Figure 3.7 shows an example for generating candidate itemsets and defining those considered large. Identifying these large itemsets is the most computationally expensive process in the Apriori algorithm. As mentioned before, Apriori does this by joining the current large set to those previously identified. This allows avoiding *a priori* new itemsets that combine elements previously discarded for not reaching minimum support. The idea is that if an itemset does not achieve minimum support, neither will do any combination including it. The example considers four different transactions (100, 200, 300, 400) including combinations of five distinct items in an initial Database D. The process steps in the example include:

- Starts with a scan on D for the identification of the size-one itemsets set ({1}, {2}, {3}, {4}, {5}).

- The subsets including the size-one itemsets per transaction are included $\bar{C}_1$.
- The support for each of the itemsets in $\bar{C}_1$ is calculated by considering the number of transactions including them, resulting in L_1.
- Considering the itemsets size one from L_1 as large (i.e. with support ≥ 2), allows the identification of the size two itemsets in C_2. $\{4\}$ is a small itemset and is not part of C_2 since its support is only one in L_1 (i.e. only appear in one transaction).
- Scanning $\bar{C}_1$ and C_2 allows the identification of the itemsets size two from the transactions results in $\bar{C}_2$.
- L_2 reflects the support for the itemsets size two from $\bar{C}_2$. The itemsets $\{1\ 2\}$ and $\{1\ 5\}$ are excluded due to their low support.
- Considering the itemsets size two from L_2 as large, allows the identification of the size three itemsets in C_3, which only includes the itemset $\{2\ 3\ 5\}$.
- L_3 shows the only itemset size three in C_3 has support value of two.
- Since there are no itemsets size four, C_4 would result in an empty set and the process ends.

Toy example using the Apriori algorithm

As with the MLP training, next are shown the results of applying the Apriori algorithm on the data from the toy example, shown in Table 3.1. Table 3.2 shows the rules resulting from using the Apriori algorithm with the *class mining* configuration[1]. Apriori generated six rules for the AND data, considering the resulting class on each combination between the variables. It is the identification of two rules as the most relevant (1 & 2) since all the rest are influenced by them. This is, if g_1 or g_2 has a value of 0, then the class will be 0 (i.e. AND=0). From the results when using the data from the OR example, it is also possible to identify the first two rules as the more relevant from the six. In this case, if g_1 or g_2 has a value of 1, then the class will be 1 (i.e. OR=1). The example with the XOR data shows that this is a harder problem that can not be described by the variables independently. Nevertheless, the resulting rules from the Apriori algorithm correctly identified the classes depending on the values taken by g_1 and g_2.

[1] *Class mining* is a parameter indicating that the process will consider only those rules having as consequence a decision the class.

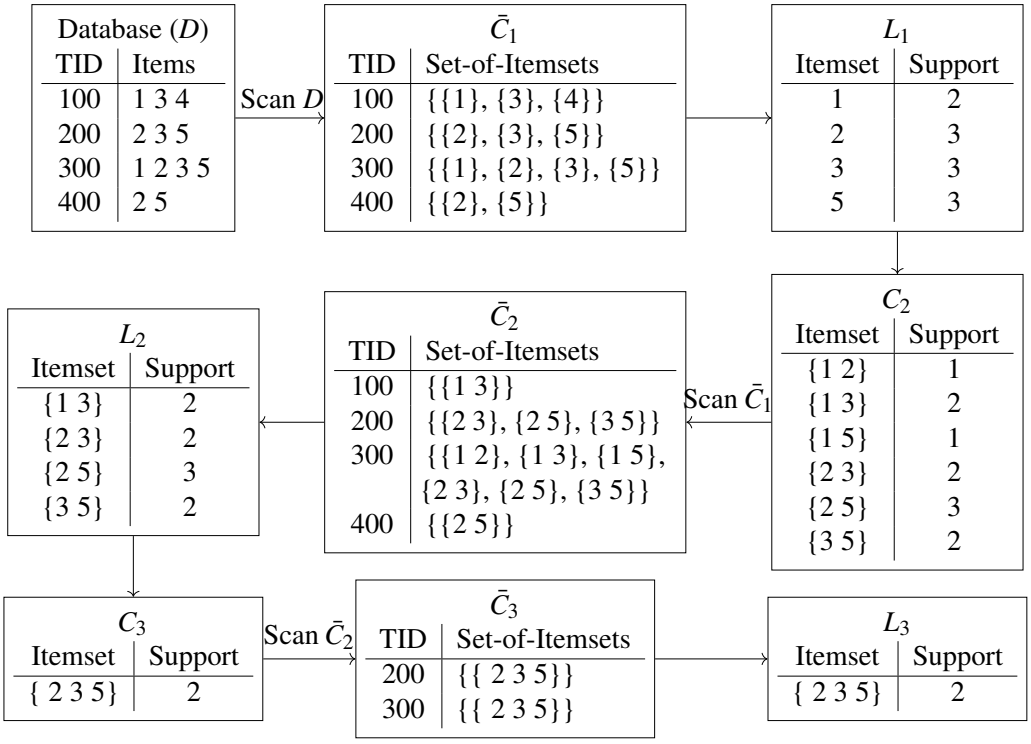


Figure 3.7 Example of definition, generation and calculation of candidate itemsets considered as large (adapted from [2]). This example considers the minimum frequency support of two transactions. The set in L_1 is the base for generating a new candidate set C_2. Iterations over the entries in $\bar{C}_1 1$ help in calculating the support of the elements in C_2. $\bar{C}_2$ results from applying a scan process on $\bar{C}_1$ and considering the itemsets in C_2. Following a similar process allows Apriori generating C_3 from L_2. Then by combining with the use of a scan process on the data in $\bar{C}_2$ and C_3 is the base for generating $\bar{C}_3$. $\bar{C}_3$ only have one largest item with size three ({2 3 5}). Finally, C_4 (not showed) results in an empty set and the process ends.

3.1.4 Decision Tree Induction

Decision Tree Induction or Decision Tree Learning generates classification rules to assign an object to a class. The Decision Tree Induction process generates the rules based on the values of the known properties from those objects. For constructing a decision tree, the process starts by defining the attribute to act as a root. This process repeats for the selection on the next attributes until finding a leaf indicating a decision on the class. The classification task using a tree begins on the root and iteratively tests at each level of the tree to define the branch to follow. This process is repeated until a leaf is found in the tree, which will determine the class. For this reason, the induction task must be able to capture meaningful relations between class

Table 3.2 Results from the use of the data in the toy example (see Table 3.1) on the Apriori algorithm. The rules correctly capture the relations between the variables and the output, even in the XOR problem.

Example	#Rule	Apriori rules	Class	Conf.
AND	1	g_1=0	C=0	(12/12, conf: 1)
	2	g_2=0	C=0	(8/8, conf: 1)
	3	g_1=0 g_2=0	C=0	(6/6, conf: 1)
	4	g_1=0 g_2=1	C=0	(6/6, conf: 1)
	5	g_1=1 g_2=1	C=1	(6/6, conf: 1)
	6	g_1=1 g_2=0	C=0	(2/2, conf: 1)
OR	1	g_2=1	C=1	(12/12, conf: 1)
	2	g_1=1	C=1	(8/8, conf: 1)
	3	g_1=0 g_2=0	C=0	(6/6, conf: 1)
	4	g_1=0 g_2=1	C=1	(6/6, conf: 1)
	5	g_1=1 g_2=1	C=1	(6/6, conf: 1)
	6	g_1=1 g_2=0	C=1	(2/2, conf: 1)
XOR	1	g_1=0 g_2=0	C=0	(6/6, conf: 1)
	2	g_1=0 g_2=1	C=1	(6/6, conf: 1)
	3	g_1=1 g_2=1	C=0	(6/6, conf: 1)
	4	g_1=1 g_2=0	C=1	(2/2, conf: 1)

and the values present in the object's attributes. Another important point in the process for tree generation by induction is that, as mentioned by Quinlan, the simpler tree is preferred when various trees are correct since it would have a better generalisation [138].

Iterative Dichotomiser 3 (ID3) algorithm

ID3, proposed by Quinlan, is a Decision Tree Induction algorithm which bases the branching decision by considering the Information Gain (IG) that an attribute represents [137]. ID3 does this by calculating and comparing the entropy[2] information from the classes against the entropy of each attribute (Equations 3.10 and 3.11 respectively).

$$I(p,n) = -\left(\frac{p}{p+n}\right)\log_2\left(\frac{p}{p+n}\right) - \left(\frac{n}{p+n}\right)\log_2\left(\frac{n}{p+n}\right) \tag{3.10}$$

[2]Entropy is a measurement proposed by Claude Shannon in Information Theory regarding the uncertainty associated with a random variable [156].

Equation 3.10 considers a subset C that contains p objects of the class P along with n from the class N. The probability to determine that an arbitrary object belongs to the class P is given by $\frac{p}{p+n}$. Similarly, the probability for the class N is given by $\frac{n}{p+n}$. With this, it is possible to calculate the expected information for a tree with the attribute A as root through the weighted average of its V possibles values given by Equation 3.11:

$$E(A) = \sum_{i=1}^{V} \left(\frac{p_i + n_i}{p+n} \right) I(p_i, n_i) \tag{3.11}$$

where the proportion of objects that belong to C_i in C gives the weight for the i^{th} branch. The values indicated by p_i and n_i refer the number of objects P and N related to the i^{th} value in the attribute A. With Equation 3.12 ID3 calculates the information gained by branching on the attribute A:

$$gain(A) = I(p,n) - E(A) \tag{3.12}$$

C4.5 Algorithm

Quinlan proposed the C4.5 algorithm as an improvement over the ID3 in 1993 [139]. The C4.5 algorithm also bases its evaluation on the information provided by the data. However, the process in C4.5 uses an IG-Ratio instead of the direct IG. One disadvantage of using only the IG is that the attributes with many outcomes get higher preference regardless of their utility [139]. The information generated by dividing a set T into n subsets, called split information, modulates the IG Ratio by Equation 3.13.

$$split\_info(X) = -\sum_{i=1}^{n} \frac{T_i}{T} \times \log_2 \left(\frac{T_i}{T} \right) \tag{3.13}$$

The IG Ratio measures the proportion of information useful from the generated split, as shown in Equation 3.14.

$$gain\_ratio(X) = \frac{gain(X)}{split\_info(X)} \tag{3.14}$$

Another improvement is the capability of pruning a generated tree based on an estimated error given by Equation 3.15:

$$N \times U_{CP}(E,N) \tag{3.15}$$

where N represents the number of examples covered in a specific leaf, with E of them classified incorrectly. With these values, C4.5 calculates the confidence limits on a binomial distribution for a given Confidence Level (CF). The upper limit ($U_{CP}(E,N)$) is then multiplied by N to get the predictive error for a specific leaf. The estimation is a pessimistic heuristic of the subtree classifying capability on unseen data. This process repeats for each leaf in a specific subtree and finally summed up the estimated errors to get the subtree's estimated predictive error. To decide if a subtree needs replacement by its branch, C4.5 considers the total number of examples in all the subtree's branches, denoted as N. Then calculates the value E, indicating the number of misclassified examples (i.e. those incorrectly classified by the selected branch). After this, C4.5 compares the estimated predictive error of the selected branch against the error of the subtree for the possible decision on replacing the subtree. If the estimated error in the branch is lower to the error from the subtree, the subtree is replaced. C4.5 uses a similar process to determine if it must remove a subtree or leaf from the tree. If the estimated predictive error without the leaf or subtree is lower than the measured estimated errors in such leaf or subtree, then it is removed.

Another pruning tool is subtree raise where C4.5 can substitute a node in the tree if its most popular subtree has a lower estimated error [179]. Raising means that a subtree at a lower level in the tree is promoted for replacing its parent subtree.

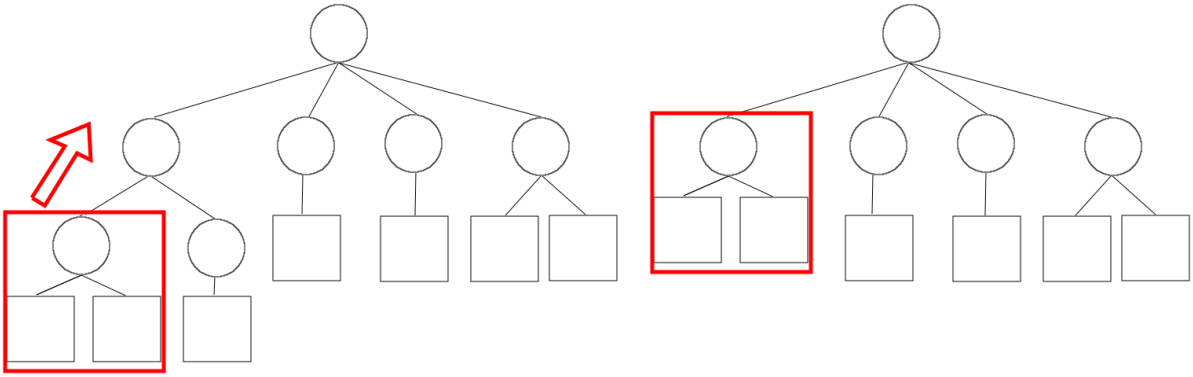


Figure 3.8 Example of the subtree raising process where a node in the tree can be substituted by its most popular subtree has a lower estimated error.

Toy example using the C4.5 algorithm

Figure 3.9 shows the resulting trees generated using C4.5 on the toy example data (see Table 3.1). These can identify the correct combinations for discriminating between Classes, even in the non-linear case of XOR.

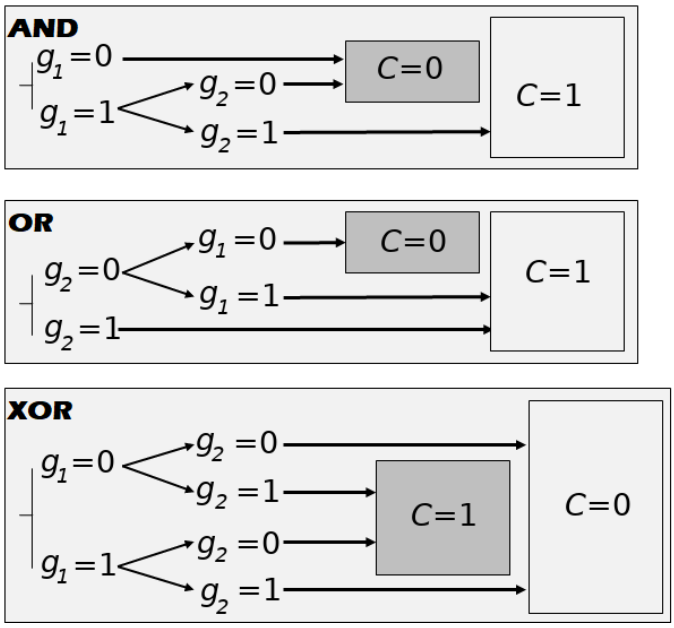


Figure 3.9 Resulting trees after the application of C4.5 on the input data of the toy example (see Table 3.1). Each tree is representing the function described by the variables g_1 and g_2, having the target class as C.

3.1.5 Attribute construction approach

Multifactor Dimensionality Reduction

Ritchie et al. proposed MDR as a non-parametric method capable of detecting high-order gene-gene and gene-environment interactions [144]. As mentioned by Moore, MDR is an approach for detecting epistasis or nonlinear gene-gene interactions though the construction of new features (i.e. models) given by the combination of the original variables [111]. By using a Cross-Validation (CV)[3] strategy, the model-free approach reduces false-positive results [64].

[3]Cross-validation: method for estimating the performance of a model. Mostly used when the data for training and testing is limited. The process makes a division of the complete dataset in k partitions. The number used for k is frequently established to 5 or 10. In this setting, the first section of the partitions is held out from the training

The method decrease dimensionality by combining the factors into high and low-risk groups. In this way, a new variable represents the data in one dimension. MDR measures the new variable capability to classify and predict a specific status by using CV and permutations.

The process begins by randomising the examples in the dataset and then assigning them into folds for the CV phase. MDR does this trying to distribute the examples by disease status as evenly as possible among the partitions. In the next step, cells in an N-dimensional space are used to represent the N attributes and their multifactor status. After that, the ratio between cases and controls in each cell is calculated. MDR defines cells as high-risk if their ratio surpasses a specific threshold T (e.g. T=1), otherwise as low-risk. This achieves reducing from N-dimensions to one with only two multifactor classes (i.e. high-risk and low-risk). This process repeats for every N-combination possible, and the best model with N factors is selected. The last step is to assess the multifactor levels that represent high-risk or low-risk. For this MDR uses the calculated ratio of cases versus controls present in the dataset as the threshold. This allows the method to adjust for an unbalanced number of cases and controls [60].

MDR uses balanced accuracy, calculated by Equation 3.16 along with the CV-consistency to select a final best model per combination size. The accuracy calculation depends on the average classification performance on the testing set (predictive accuracy). On the other hand, the CV-consistency indicates in how many folds of the CV process the final best model was selected.

$$\text{balanced accuracy} = \frac{\text{sensitivity} + \text{specificity}}{2} \tag{3.16}$$

Toy example using the MDR algorithm

The models generated by MDR on the toy example data (see Table 3.1) are shown in Figure 3.10. MDR is capable of identifying the combinations that give the correct Class as a result.

3.2 Proposed machine learning methodology

In Figure 3.11, we present our methodology for identifying the best combinations of variables and class. In the first step, the data is processed by three different algorithms: Apriori, C4.5 and MDR. Figure 3.12, Figure 3.13 and Figure 3.15 show examples of the process for each algorithm. After creating the rules, trees and models from the algorithms, assessing the most

process, which uses the rest of the partitions. After training takes place, the held partition is used for testing. This process is repeated for each partition, and finally, the performance is calculated as the mean of the evaluations using the test partitions [64].

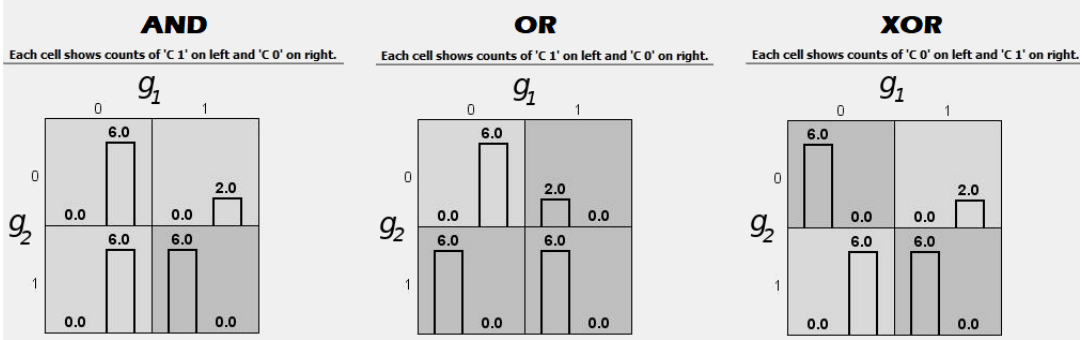


Figure 3.10 Two-variable models after the application of MDR on the input data of the toy example. Each model is representing the "risk" associated with each function. Therefore, each Class is identified by different cell colour. The models describe the interaction between the variables g_1 and g_2, having the target class as C.

important variables in them is the next step. For this, the methodology considers the consistency of their inclusion in the algorithms' results and their rank or CV classification performance. The use of MLP further improves assessing the importance of the combination of variables by evaluating classification performance as shown in Figure 3.16. After identifying the most relevant variables, the methodology uses them as input in the ID3 algorithm for the suggestion of specific combinations as hypotheses. In this way, each branch in the ID3-inducted decision tree acts as a hypothesis in a traditional statistical method. Next is presented the methodology process in detail. For details on the algorithms used, see Subsections 3.1.1-3.1.5.

The process in the methodology starts with the data entry indicating the input data. The first step is data encoding, involving data cleaning and the selection of the data representation. Association rules generation is through the Apriori algorithm. Its use has proved to help in identifying associations among genetic features [148]. This approach is computationally expensive since it includes testing all the combinations among the variables present in the dataset in the worst case. For this reason, it is necessary to limit the variables to be considered in the dataset. Furthermore, the computation burden might increase considering a low confidence level (i.e. 0.1) that might be necessary when working with an imbalanced dataset[4]. This might allow the inclusion of rules representing the less represented class in the dataset. Once the rules had been generated, the next step is selecting the top ten rules for each class for further analysis. The consideration of using only the first ten rules came from the fact that previous research showed that the first rules are the ones that capture the most relevant interactions [148]. This

[4]In supervising learning an imbalanced dataset is such that has an over-representation of elements corresponding to some target while having fewer elements of other targets.

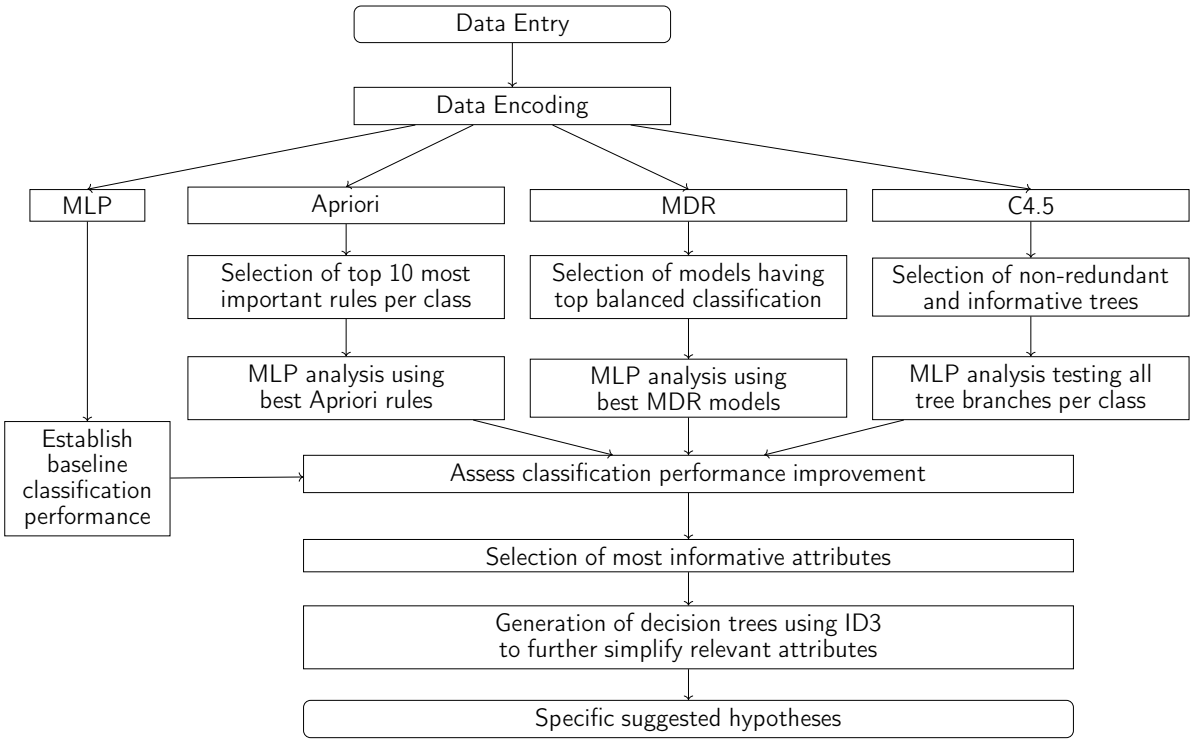


Figure 3.11 Methodology for the identification of the best combinations.

allows assessing the variables present in each rule in the next step, considering the classification performance when included in the combinations from the rules. For this, the MLP classification task only includes the variables in the rules and not their values.

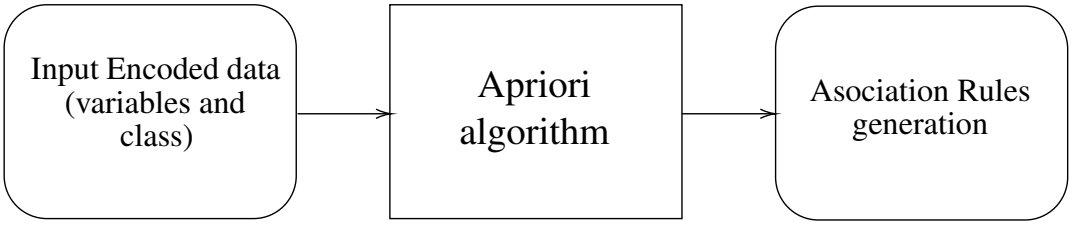


Figure 3.12 Example of the use of the Apriori algorithm for the generation of association rules. The Apriori algorithm uses the variables along with the classes as input for Association Rule Learning. By using rule mining, Apriori allows identifying the most relevant rules related to the classes.

With the C4.5 algorithm, fifty-one decision trees are generated by modifying its pruning parameter CF with increments of 0.01 in the range [0.01-0.51]. This range represents the space of possible values for the CF while the increment value reflects an exploration considering small steps. The C4.5 decision tree induction process evaluates the variables' importance by considering the information gain that each of them provides. After generating the trees, the next step is to identify the informative and distinct ones as shown in Figure 3.14. Uninformative trees are those that do not include any test on the variables (i.e. trees with size 1). Those combinations of variables identified in the trees are used in the next step for assessing their classification performance. These combinations include the variables present in each branch of the trees. As with the Apriori algorithm, the process ignores the values of the variables in the trees for the MLP classification task.

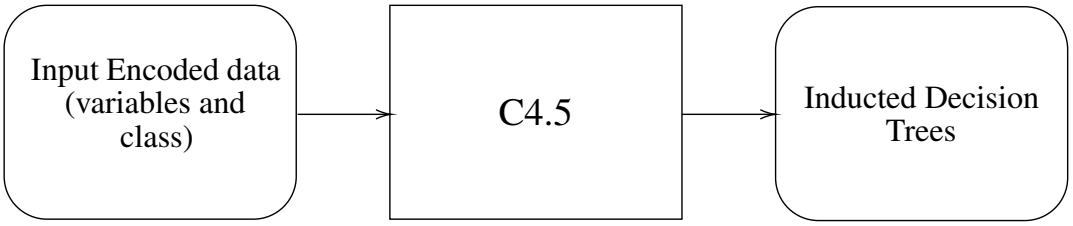


Figure 3.13 Example of the process for decision tree induction using the C4.5 algorithm. C4.5 uses the variables and classes as input for Decision Tree Induction considering different CF values produce.

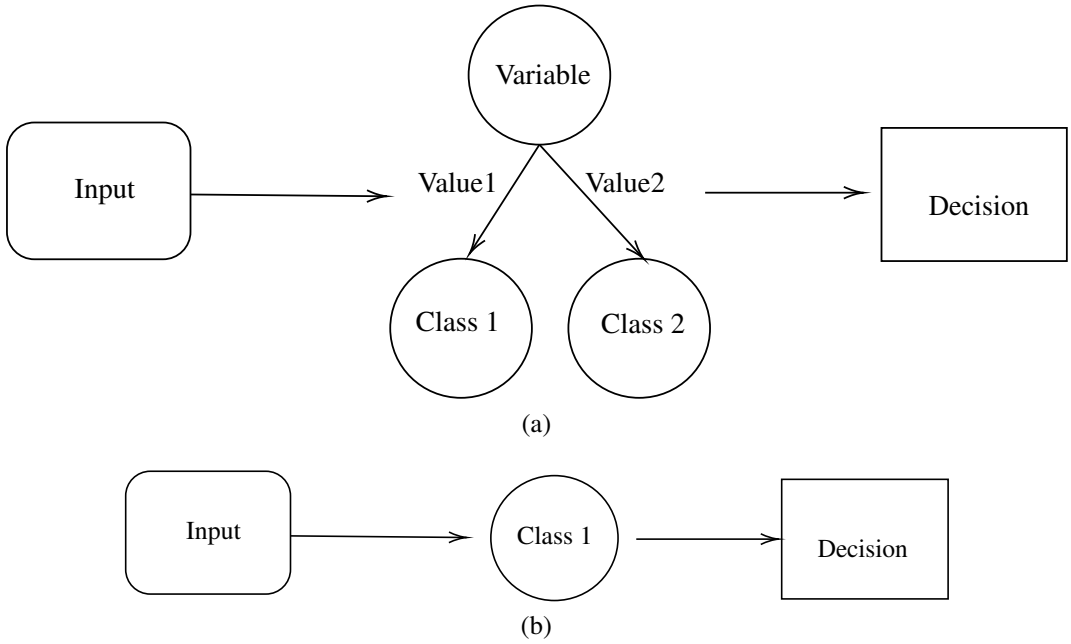


Figure 3.14 Example the definition of informative tree: a)shows the example of an *informative* inducted decision tree of size one having one test determining the classification; b) shows that when the tree is *uninformative* (i.e. size is zero), the classification result is the determination of a class without testing.

The third algorithm is MDR which can identify a subset of the most informative variables. After identifying them, their combinations allow constructing new variables that are tested in a later classification task. In the next step, MDR selects the best model considering the balanced accuracy and CV-consistency. As with the Apriori and C4.5 algorithms, the variables included in the models are later used as input for testing in the MLP classification phase.

Figure 3.15 Example of the use of MDR for generating interaction models (i.e. models in that capture interacting relationships among the variables). MDR uses the variables and the classes as input for model learning. The exploration includes models from one to six variables, considering their influence on the classification capability.

The use of the MLP in the methodology is in a twofold manner. First, for calculating the classification performance on using all the variables in the dataset to define a baseline value, as shown in Figure 3.16a. In this sense, MLP is used for calculating the CV-accuracy on the complete dataset. Secondly, the MLP is used for calculating the CV-classification performance on using only specific variables from the dataset, shown in Figure 3.16a. These variables are defined by their presence in the Apriori-rules, C4.5 decision tree branches or MDR-models. The comparison of the classification performance from each specific combination of variables against the baseline is used as an indicator on the relevance of the selected variables.

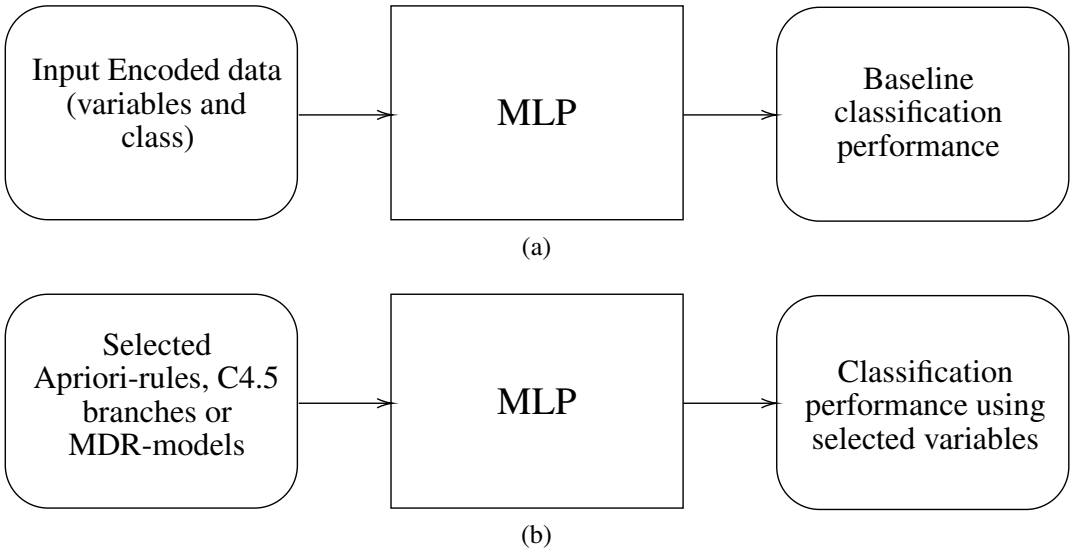


Figure 3.16 Example of the use of MLP for generating interaction models: a) shows the initial process for calculating the baseline classification performance using all the available variables; b) shows the process for evaluating specific combinations of variables according to those selected in the Apriori-rules, C4.5-tree branches or MDR-models.

The results from the three approaches along with the variable combinations assessment made by the MLP serve to measure the importance of the variables. For this, the methodology calculates an accumulation for each variable present in the results of the four previous methods. The definition on the points was done considering an empirical approach based on: i) the rules were considered with a lower value since the methodology only explores a small, although informative, fraction of all the possible combinations; ii) since the search of interactions is more informative with MDR than with the Apriori algorithm, its results are assigned more value; and iii) finally, the results of C4.5 receive the highest value since it was applied without restrictions

(i.e. without limiting the number of values to be included in the trees). The points assigned to each variable per algorithm were determined considering:

1. Points assigned for the Apriori algorithm:
   (a) Two points for each time a variable is present in one of the selected rules.
   (b) The proportion of 20 points considering variables in selected rules that improve the MLP-baseline classification performance in relation to all the selected rules.
2. Points assigned for the C4.5 algorithm:
   (a) Six points are added to each variable considering the tree-branches in which it is included.
   (b) Considering the variable included in the trees, three additional proportions are considered:
      i. The proportion of ten points considering in how many of the distinct trees each variable was included.
      ii. The proportion of ten points considering the rank that each variable holds in the distinct trees. This value is calculated using Equation 3.17. The proportion depends on the average level at which the variable is found.
      iii. A proportion of 30 points by considering how many of the classification-branches that improve the baseline classification include the variable.
3. Points assigned for the MDR algorithm:
   (a) Eight points to each variable present in each of the cells from the best models. This is similar to finding the variable in a decision tree branch.
   (b) The proportion of 20 points considering in how many of the selected models with combinations improving the baseline performance appears when considering all the possible combinations on the selected models.

In the case of the results from the C4.5 algorithm, the points for the rank are distributed among the variables included in the trees, with a value that decreases as proportional to the average rank. Equation 3.17 determines the proportion per each variable.

$$\text{variable}_{rank-points} = \frac{(\#\text{ variables included in the trees}+1)-(\text{average variable}_{rank})}{\#\text{variables included in the trees}} \times 10 \quad (3.17)$$

The last step makes use of the ID3 considering the variables identified as more relevant for constructing decision trees without pruning, as shown in Figure 3.17. Since the selection process might deliver a small number of variables, ID3 was preferred for the induction of the final trees. ID3 was selected considering that it includes neither pruning nor other processes that can end in an uninformative tree. ID3 suggests specific combinations for further statistical and biological assessment considering each branch in the inducted decision tree. The advantage of using ID3 in the last step is that it reduces the number of combinations for testing as hypotheses. Finally, these hypotheses are outputted as the finals results of the methodology.

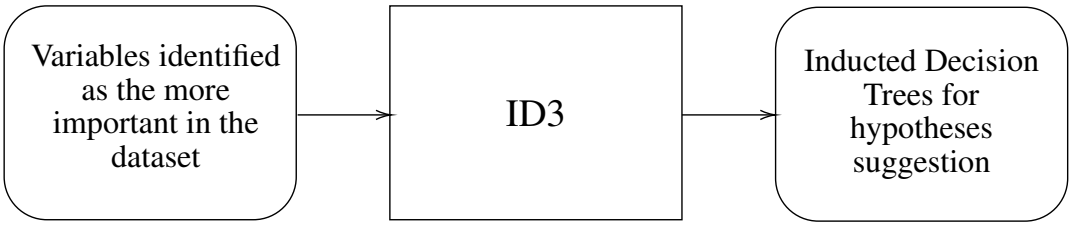


Figure 3.17 Example of the use of ID3 for the definition of the final inducted decision tree. The resulting hypotheses from the methodology are taken from each branch of this tree.

Chapter 4

Results and discussion

After introducing the methodology in Chapter 3, this chapter presents the results of its application to the Human Immunodeficiency Virus (HIV)-viral infectivity factor (vif) dataset presented in Chapter 1. This process allowed the identification of relevant variables and the suggestion of some meaningful combinations. It also presents the results from the statistical assessment on the hypotheses suggested by the methodology. Finally, it gives a discussion about the results.

4.1 Steps for Pattern recognition

The goal of pattern recognition is to achieve the best description of the objects of interest. An appropriate description of the objects can improve their correct discrimination or identification. The steps required for defining the descriptors or features used in this thesis are presented next and are also shown in Figure 1.4. All the following steps were applied per each clinical descriptor or *output* (i.e. *CD4Ini*, *CD4Hist*, *VLIni* and *VLHist*).

In applying our methodology, as shown in Figure 3.11, the step *Data Encoding* involved the analysis of the original dataset. This process, summarised in Annexe B, allowed the identification of the binary encoding and 17 variables as the best representation, as shown in Table 1.1. Further analysis allowed the identification of the variable *APOBEC-1* as non-informative since all the sequences had a value of "1" in it. Therefore, this variable was removed for the rest of the experiments.

Once the dataset for experimentation has been defined with sixteen binary variables, the next phases involve generating association rules, models and decision trees by using Apriori, C4.5, Iterative Dichotomiser 3 (ID3) and Multifactor Dimensionality Reduction (MDR) on

the codified dataset. The process also includes calculating the Multi-Layer Perceptron (MLP) baseline-accuracy performances. The Weka's implementations of the Apriori algorithm, ID3, J48 (for C4.5) and MLP by Hall et al. were used for this part of the experimentation [61]. The MDR implementation by Hanh et al. was used for the model generation [60]. This implementation includes the attribute assessment proposed by Jakulin and Bratko that helps in identifying a subset of the most informative variables [74]. MDR then combines these attributes for constructing new ones, which are the inputs in the classification task. The framework also provides information on the relations between attributes, which is useful for their interpretation. These relations can reflect redundancy, synergy or a mix of both. A redundant relation exists when the variables present correlation (i.e. they have the same information when considering the discrimination between classes). On the other hand, a relation reflects synergy when it increases the information that helps to discriminate the classes. These implementations were preferred since they are proven tools for data mining. In the case of MDR, it had shown being useful in the search for genetic associations [109, 98]. The use of these implementations has the advantage of using correct and prove developments. On the other hand, using them limits the possibility of adding additional improvements that might help in the search.

Our methodology uses as input the combinations of all the available variables and each output for the MLP training process. In this way, we define the MLP baseline classification performance per output. The baseline performance reflects the 10-Cross-Validation (CV)-accuracy on the testing set. The parameters for the training processes with MLP are shown in Table 4.1. The candidate combinations of variables from rules, models and decision trees were then compared with the baseline performance considering each output. MLP-baseline performance per output was 76.62% for Initial Cluster of Differentiation 4 (CD4)+ T cell lymphocytes (CD4Ini), 71.43% for historic CD4+ T cell lymphocytes (CD4Hist), 64.94% for Initial viral load (VLIni) and 55.84% for historic viral load (VLHist). The baseline performance for CD4Ini and CD4Hist reflects the data imbalance, where the models are influenced by the more represented class, as shown in Table 1.2. In the case of VLIni and VLHist, the baseline reflects the difficulty in discriminating between the classes, possibly showing that more examples are needed. Nevertheless, these baseline performances help in identifying combinations of variables that improve class discrimination.

Table 4.1 Parameters description for the MLP training. These describe both the definition for the baseline classification performance and the evaluation on the Apriori, C4.5 and MDR results. The value *iv* describes the number of *input variables* from the selected rules, decision tree branches and models. The numbers for the architecture describes the number of neurons in the input, hidden and output layers, respectively.

Parameter	For baseline	For testing on results
Architecture	[16 , 8, 2]	[iv, $\frac{iv+2}{2}$, 2]
Learning rate (α)	0.3	0.3
Momentum	0.2	0.2
Epochs	500	500

4.1.1 Apriori algorithm process

The process for association rules generation used a confidence value of 0.1 and *rule mining* with the class as the *head* (i.e. the consequence of the rule). Setting a low confidence value increases the probability of generating rules for the less represented class. The total generated rules per output were 135,598 for CD4Ini, 138,062 for CD4Hist, 112,926 for VLIni and 124,966 for VLHist. Table 4.2 shows the rules per output and class. The methodology selects the ten Apriori-generated rules per class and output for further analysis. These rules are ones with the highest confidence values for every class on each output.

Table 4.2 Number of rules generated by Apriori per output.

Output	$\geq$500 CD4+ T cells/μL (Class 0)	$<$500 CD4+ T cells/μL (Class 1)	Output total
CD4Ini	5,695	129,903	135,598
CD4Hist	4,911	133,151	138,062

Output	$<$10K cp/mL (Class 0)	$\geq$10K cp/mL (Class 1)	Output total
VLIni	73,215	124,966	124,966
VLHist	27,671	85,255	112,926

Table 4.3 shows the selected rules for the clinical status on CD4+ T cell counts and their confidence values, which can be in the range (0.0-1.0]. The selected rules for CD4Ini with the Class=1 ($<$ 500 cells per μL) were the ones with the higher confidence values. All of them had a confidence of 1.0 since the consequence of having the combination described by the rule was CD4Ini=1 in all the cases. For example, the rule *APOBEC-2*$^{\text{Mut}}$ and *APOBEC-4*$^{\text{Cons}}$ was present in 15 examples and all had CD4Ini=1 as consequence. On the other hand, those for the Class=0 ($\geq$ 500 cells per μL) are present only after the rule number 129,836. This reflects the fact that most of the sequences in the imbalanced dataset are from Class 1. All the selected rules

having the Class=1 as head included the definitions[1] of *APOBEC-2*$^{\text{Mut}}$ and *APOBEC-4*$^{\text{Cons}}$ in them. On the other hand, all the selected rules with Class=0 as head included the definition (i.e. specific variables-values combination) of *BCbox-3*$^{\text{Mut}}$ while *APOBEC-3*$^{\text{Cons}}$ was present in most of them.

As in the case of CD4Ini, in CD4Hist the values in confidence for the Class=1 were the highest. In this case, the definitions of *APOBEC-2*$^{\text{Mut}}$, *APOBEC-3*$^{\text{Cons}}$ and *APOBEC-7*$^{\text{Cons}}$ were present in the top ten rules having Class=1 as the head. In the case of the selected rules having the Class=0 as the head, all of them included the definitions of *APOBEC-2*$^{\text{Cons}}$, *APOBEC-3*$^{\text{Cons}}$, *BCbox-2*$^{\text{Mut}}$ and *Cul5-3*$^{\text{Mut}}$. The definition of *APOBEC-3*$^{\text{Cons}}$ was the only one that appeared in the top ten selected rules for both classes. These rules suggested that the segment described by *APOBEC-3* have an important role in the levels of the CD4+ T cells.

Table 4.4 shows the selected rules for the clinical status considering viral loads and their confidence values. VLIni also presented higher confidence for the rules having Class=1 ($\geq$10K viral copies) as the head. In this case, the definitions of *APOBEC-3*$^{\text{Cons}}$, *APOBEC-7*$^{\text{Cons}}$, *BCbox-3*$^{\text{Cons}}$ and *CBFb-2*$^{\text{Mut}}$ were present in all the rules related to Class=1. On the other hand, the definitions *APOBEC-2*$^{\text{Cons}}$, *BCbox-1*$^{\text{Cons}}$, *BCbox-2*$^{\text{Cons}}$, *Cul5-3*$^{\text{Mut}}$ and *NLIS*$^{\text{Cons}}$ were present in all the selected rules for Class=0 ($<$10K viral copies).

The selected rules for VLHist were the only ones where the Class=0 had the highest confidence. This is explained for the fact that in this output the most represented class is "0" (see Table 1.2). All of these rules included the definitions of *APOBEC-2*$^{\text{Cons}}$, *BCbox-2*$^{\text{Cons}}$ and *NLIS*$^{\text{Cons}}$. In the case of the rules selected with Class=1 as head, all of them included the definitions of *APOBEC-2*$^{\text{Mut}}$, *APOBEC-4*$^{\text{Mut}}$, *APOBEC-8*$^{\text{Cons}}$ and *BCbox-1*$^{\text{Cons}}$.

[1]Here, *definition* expresses some specific variable having a specific value. For example, *APOBEC-2*$^{\text{Mut}}$ describes the variable *APOBEC-2* with the specific value of "1" (i.e. mutated), as described in section 1.4.

Table 4.3 Apriori top ten class-mined rules per class with the outputs CD4Ini and CD4Hist, for a confidence value of 0.1.

	#Rule	Apriori rules							Class	Conf.
Initial CD4 T lymphocytes (CD4Ini)	1	APOBEC-2^{Mut}	APOBEC-4^{Cons}						CD4Ini=1	(15/15, conf: 1)
	2	APOBEC-2^{Mut}	APOBEC-4^{Cons}	CBFb-1^{Cons}					CD4Ini=1	(15/15, conf: 1)
	3	APOBEC-2^{Mut}	APOBEC-4^{Cons}	Cul5-1^{Cons}					CD4Ini=1	(15/15, conf: 1)
	4	APOBEC-2^{Mut}	APOBEC-4^{Cons}	Cul5-2^{Cons}					CD4Ini=1	(15/15, conf: 1)
	5	APOBEC-2^{Mut}	APOBEC-4^{Cons}	CBFb-1^{Cons}	Cul5-1^{Cons}				CD4Ini=1	(15/15, conf: 1)
	6	APOBEC-2^{Mut}	APOBEC-4^{Cons}	CBFb-1^{Cons}	Cul5-2^{Cons}				CD4Ini=1	(15/15, conf: 1)
	7	APOBEC-2^{Mut}	APOBEC-4^{Cons}	Cul5-1^{Cons}	Cul5-2^{Cons}				CD4Ini=1	(15/15, conf: 1)
	8	APOBEC-2^{Mut}	APOBEC-4^{Cons}	CBFb-1^{Cons}	Cul5-1^{Cons}	Cul5-2^{Cons}			CD4Ini=1	(15/15, conf: 1)
	9	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Cons}					CD4Ini=1	(14/14, conf: 1)
	10	APOBEC-2^{Mut}	APOBEC-4^{Cons}	APOBEC-5^{Cons}					CD4Ini=1	(14/14, conf: 1)
	129836	APOBEC-3^{Cons}	BCbox-3^{Mut}						CD4Ini=0	(8/14, conf: 0.57)
	129837	APOBEC-8^{Cons}	BCbox-3^{Mut}						CD4Ini=0	(8/14, conf: 0.57)
	129838	APOBEC-3^{Cons}	APOBEC-5^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129839	APOBEC-3^{Cons}	APOBEC-6^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129840	APOBEC-3^{Cons}	APOBEC-8^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129841	APOBEC-3^{Cons}	CBFb-1^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129842	APOBEC-3^{Cons}	Cul5-1^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129843	APOBEC-3^{Cons}	Cul5-2^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129844	APOBEC-3^{Cons}	BCbox-1^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
	129845	APOBEC-5^{Cons}	APOBEC-8^{Cons}	BCbox-3^{Mut}					CD4Ini=0	(8/14, conf: 0.57)
Historical CD4+ T lymphocytes (CD4Hist)	1	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}					CD4Hist=1	(21/21, conf: 1)
	2	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}				CD4Hist=1	(21/21, conf: 1)
	3	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	Cul5-1^{Cons}				CD4Hist=1	(21/21, conf: 1)
	4	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	Cul5-2^{Cons}				CD4Hist=1	(21/21, conf: 1)
	5	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	BCbox-1^{Cons}				CD4Hist=1	(21/21, conf: 1)
	6	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}	Cul5-1^{Cons}			CD4Hist=1	(21/21, conf: 1)
	7	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}	Cul5-2^{Cons}			CD4Hist=1	(21/21, conf: 1)
	8	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}	BCbox-1^{Cons}			CD4Hist=1	(21/21, conf: 1)
	9	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	Cul5-1^{Cons}	Cul5-2^{Cons}			CD4Hist=1	(21/21, conf: 1)
	10	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	Cul5-1^{Cons}	BCbox-1^{Cons}			CD4Hist=1	(21/21, conf: 1)
	133152	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-6^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}			CD4Hist=0	(8/17, conf: 0.47)
	133153	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-6^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}		CD4Hist=0	(8/17, conf: 0.47)
	133154	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-6^{Cons}	CBFb-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}		CD4Hist=0	(8/17, conf: 0.47)
	133155	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-6^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}		CD4Hist=0	(8/17, conf: 0.47)
	133156	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-6^{Cons}	CBFb-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}	CD4Hist=0	(8/17, conf: 0.47)
	133157	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-6^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}	CD4Hist=0	(8/17, conf: 0.47)
	133158	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-6^{Cons}	CBFb-1^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}	CD4Hist=0	(8/17, conf: 0.47)
	133159	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-6^{Cons}	CBFb-1^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}	CD4Hist=0	(8/17, conf: 0.47)
	133160	APOBEC-2^{Cons}	APOBEC-3^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}					CD4Hist=0	(8/18, conf: 0.44)
	133161	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	Cul5-3^{Mut}	BCbox-2^{Mut}				CD4Hist=0	(8/18, conf: 0.44)

Table 4.4 Apriori top ten class-mined rules per class with the outputs VLIni and VLHist, considering a confidence value of 0.1.

	#Rule	Apriori rules							Class	Conf.
Initia viral load (VLIni)	1	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}				VLIni=1	(8/8, conf: 1)
	2	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	3	APOBEC-3^{Cons}	APOBEC-6^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	4	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	5	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	Cul5-1^{Cons}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	6	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	Cul5-2^{Cons}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	7	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	BCbox-1^{Cons}	BCbox-3^{Cons}			VLIni=1	(8/8, conf: 1)
	8	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-6^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}		VLIni=1	(8/8, conf: 1)
	9	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-7^{Cons}	CBFb-1^{Cons}	CBFb-2^{Mut}	BCbox-3^{Cons}		VLIni=1	(8/8, conf: 1)
	10	APOBEC-3^{Cons}	APOBEC-5^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Mut}	Cul5-1^{Cons}	BCbox-3^{Cons}		VLIni=1	(8/8, conf: 1)
	16016	APOBEC-2^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}			VLIni=0	(10/13, conf: 0.77)
	16185	APOBEC-2^{Cons}	APOBEC-3^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}		VLIni=0	(10/13, conf: 0.77)
	16186	APOBEC-2^{Cons}	APOBEC-5^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}		VLIni=0	(10/13, conf: 0.77)
	16187	APOBEC-2^{Cons}	APOBEC-6^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}		VLIni=0	(10/13, conf: 0.77)
	16188	APOBEC-2^{Cons}	NLIS^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}		VLIni=0	(10/13, conf: 0.77)
	16189	APOBEC-2^{Cons}	NLIS^{Cons}	Cul5-2^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}		VLIni=0	(10/13, conf: 0.77)
	16422	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-5^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}	VLIni=0	(10/13, conf: 0.77)
	16423	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-6^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}	VLIni=0	(10/13, conf: 0.77)
	16424	APOBEC-2^{Cons}	APOBEC-3^{Cons}	NLIS^{Cons}	Cul5-1^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}	VLIni=0	(10/13, conf: 0.77)
	16425	APOBEC-2^{Cons}	APOBEC-3^{Cons}	NLIS^{Cons}	Cul5-2^{Cons}	Cul5-3^{Mut}	BCbox-1^{Cons}	BCbox-2^{Cons}	VLIni=0	(10/13, conf: 0.77)
Historical viral load (VLHist)	1	APOBEC-2^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Cons}	NLIS^{Cons}	BCbox-2^{Cons}			VLHist=0	(11/13, conf: 0.85)
	2	APOBEC-2^{Cons}	APOBEC-7^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-2^{Cons}			VLHist=0	(11/13, conf: 0.85)
	3	APOBEC-2^{Cons}	CBFb-1^{Cons}	CBFb-2^{Cons}	NLIS^{Cons}	BCbox-2^{Cons}			VLHist=0	(11/13, conf: 0.85)
	4	APOBEC-2^{Cons}	CBFb-1^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-2^{Cons}			VLHist=0	(11/13, conf: 0.85)
	5	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Cons}	NLIS^{Cons}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	6	APOBEC-2^{Cons}	APOBEC-3^{Cons}	APOBEC-7^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	7	APOBEC-2^{Cons}	APOBEC-3^{Cons}	CBFb-1^{Cons}	CBFb-2^{Cons}	NLIS^{Cons}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	8	APOBEC-2^{Cons}	APOBEC-3^{Cons}	CBFb-1^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	9	APOBEC-2^{Cons}	APOBEC-5^{Cons}	APOBEC-7^{Cons}	CBFb-2^{Cons}	NLIS^{Cons}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	10	APOBEC-2^{Cons}	APOBEC-5^{Cons}	APOBEC-7^{Cons}	NLIS^{Cons}	Cul5-3^{Mut}	BCbox-2^{Cons}		VLHist=0	(11/13, conf: 0.85)
	10916	APOBEC-2^{Mut}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	BCbox-1^{Cons}				VLHist=1	(8/11, conf: 0.73)
	10938	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	BCbox-1^{Cons}			VLHist=1	(8/11, conf: 0.73)
	10939	APOBEC-2^{Mut}	APOBEC-4^{Mut}	APOBEC-5^{Cons}	APOBEC-8^{Cons}	BCbox-1^{Cons}			VLHist=1	(8/11, conf: 0.73)
	10940	APOBEC-2^{Mut}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	CBFb-1^{Cons}	BCbox-1^{Cons}			VLHist=1	(8/11, conf: 0.73)
	10941	APOBEC-2^{Mut}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	Cul5-1^{Cons}	BCbox-1^{Cons}			VLHist=1	(8/11, conf: 0.73)
	10942	APOBEC-2^{Mut}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	Cul5-2^{Cons}	BCbox-1^{Cons}			VLHist=1	(8/11, conf: 0.73)
	11060	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Mut}	APOBEC-5^{Cons}	APOBEC-8^{Cons}	BCbox-1^{Cons}		VLHist=1	(8/11, conf: 0.73)
	11061	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	CBFb-1^{Cons}	BCbox-1^{Cons}		VLHist=1	(8/11, conf: 0.73)
	11062	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	Cul5-1^{Cons}	BCbox-1^{Cons}		VLHist=1	(8/11, conf: 0.73)
	11063	APOBEC-2^{Mut}	APOBEC-3^{Cons}	APOBEC-4^{Mut}	APOBEC-8^{Cons}	Cul5-2^{Cons}	BCbox-1^{Cons}		VLHist=1	(8/11, conf: 0.73)

Table 4.5 shows a summary of the variables included in the selected Apririo-rules per output. In the case of CD4Ini and CD4Hist the selected rules considered eleven distinct variables in each of them. On the other hand, fourteen different variables were included in each of the VLIni and VLHist outputs. The results show that determining the most relevant variables require further analysis.

Table 4.5 Variables included in the selected Apriori-rules, shown with a value 1 per variable and output.

Output	**APOBEC-2**	**APOBEC-3**	**APOBEC-4**	**APOBEC-5**	**APOBEC-6**	**APOBEC-7**	**APOBEC-8**	**CBFb-1**	**CBFb-2**	**NLIS**	**Cul5-1**	**Cul5-2**	**Cul5-3**	**BCbox-1**	**BCbox-2**	**BCbox-3**
CD4Ini	1	1	1	1	1		1	1			1	1		1		1
CD4Hist	1	1		1	1	1		1			1	1	1	1	1	
VLIni	1	1		1	1	1		1	1	1	1	1	1	1	1	1
VLHist	1	1	1	1		1	1	1	1	1	1	1	1	1	1	

4.1.2 MDR algorithm process

MDR was used for the search of models including from one to six variables. The configuration for MDR included using 10-CV, the assignment of the most represented class on tie cells and the exploration of the top 500 models. The MDR process examines the information increase resulting from combining variables using a similar measurement as the entropy from the ID3 and C4.5 algorithms. This helps in the detection of relevant interactions while allowing to reduce the number of combinations of variables to explore. After creating the models, the next step was selecting the best ones considering the balanced accuracy values on the testing set, using Equation 3.16. The performance ranges for the generated models were: [52.05%-59.58%] for CD4Ini, [44.98%-56.66%] for CD4Hist, [47.19%-61.73%] for VLIni and [44.7%-62.88%] for VLHist. The classification performance can be considered as low, especially for CD4Ini and CD4Hist. As with the MLP training process, these results reflect the difficulty in learning from imbalanced datasets. The performance is lower than that of the baseline MLP since MDR used a more strict measure, balanced accuracy. Nevertheless, these models capture information on the interaction among variables that can help in their assessment. For this reason, the models with the three highest values were selected for further analysis, considering only models achieving over 50% in balanced accuracy. With this last consideration, all those models having worst performance than using a random approach are discarded. On the other hand, the reason for limiting to at most three selected models is that the models generally are composed by the same base of variables. This is similar to the presence of the same variables in the most relevant Apriori-rules.

Table 4.6 shows the MDR resulting models per output. In the case of CD4Ini, the balanced accuracy on the testing set from the best models was 59.58%, 56.45% and 56.45% considering one, two and six variables respectively. The resulting model with two variables for CD4Ini is shown in Figure 4.1 as an example. It shows that most of the sequences are identified by having the variables *BCbox-3* and *Cul5-2* with the value of *Cons* (i.e. with "0"). The rest of the learned models are not shown since the ones having more than three variables are difficult to fit. Nevertheless, all of them were analysed in the remaining steps of the methodology. *BCbox-3* was present in all the models, suggesting a relation between the variable and the clinical output. The best models generated for CD4Hist were those having from one to three variables with a balanced accuracy of 56.66%, 55.84% and 55.17% respectively. The variable *APOBEC-2* was present in all the models, while *APOBEC-3* was in two. The best MDR for VLIni achieved a balanced accuracy of 60.46%, 61.73% and 57.65% with three, four and five variables, respectively. The best models included the variables *APOBEC-2*, *BCbox-2* and *NLIS*.

In the case of the output VLHist, only two models were selected as the best since the rest did not reach over 50% in balanced accuracy. The best models had one and two variables with a performance of 62.88% and 57.58% respectively.

Table 4.6 Generated MDR models per output. A boldface typeset shows the ones selected as best by considering the balanced accuracy and CV-consistency.

Output	Model	Model detail	% Balanced Acc. Training	% Balanced Acc. Testing	CV-Consistency
CD4Ini	**1**	**BCbox-3**	**69.62**	**59.58**	**8/10**
	2	**Cul5-2, BCbox-3**	**73.00**	**56.45**	**6/10**
	3	Cul5-3, BCbox-2, BCbox-3	76.54	52.05	5/10
	4	APOBEC-4, Cul5-3, BCbox-2, BCbox-3	80.34	54.66	4/10
	5	APOBEC-7, CBFb-2, Cul5-3, BCbox-2, BCbox-3	84.85	56.30	6/10
	6	**APOBEC-4, APOBEC-7, CBFb-2, Cul5-3, BCbox-2, BCbox-3**	**87.25**	**56.45**	**5/10**
CD4Hist	**1**	**APOBEC-2**	**65.38**	**56.66**	**6/10**
	2	**APOBEC-2, APOBEC-3**	**71.02**	**55.84**	**6/10**
	3	**APOBEC-2, APOBEC-3, Cul5-3**	**75.36**	**55.17**	**5/10**
	4	APOBEC-2, CBFb-2, Cul5-3, BCbox-2	79.92	44.98	3/10
	5	APOBEC-2, APOBEC-4, Cul5-3, BCbox-2, BCbox-3	84.63	47.75	6/10
	6	APOBEC-2, APOBEC-7, CBFb-2, Cul5-3, BCbox-2, BCbox-3	87.95	52.51	3/10
VLIni	1	APOBEC-2	62.39	47.19	7/10
	2	APOBEC-2, BCbox-1	67.66	53.32	8/10
	3	**APOBEC-2, NLIS, BCbox-2**	**71.88**	**60.46**	**8/10**
	4	**APOBEC-2, APOBEC-7, NLIS, BCbox-2**	**75.85**	**61.73**	**6/10**
	5	**APOBEC-2, APOBEC-7, NLIS, BCbox-1, BCbox-2**	**78.71**	**57.65**	**5/10**
	6	APOBEC-2, APOBEC-4, NLIS, Cul5-3, BCbox-1, BCbox-2	81.07	52.55	4/10
VLHist	**1**	**NLIS**	**62.88**	**62.88**	**10/10**
	2	**NLIS, Cul5-1**	**65.70**	**57.58**	**7/10**
	3	APOBEC-7, NLIS, BCbox-3	68.77	50.00	7/10
	4	APOBEC-2, APOBEC-7, APOBEC-8, CBFb-2	71.93	45.08	3/10
	5	APOBEC-2, APOBEC-7, APOBEC-8, NLIS, BCbox-2	74.96	44.70	3/10
	6	APOBEC-2, APOBEC-4, CBFb-2, NLIS, BCbox-2, BCbox-3	77.95	46.97	4/10

The variables suggested as more relevant in by MDR are shown in Table 4.7. The six models for CD4Ini and VLIni both included seven different variables, while for CD4Hist and VLHist had eight and nine respectively. Although these results show that MDR includes fewer variables than Apriori, further analysis is required since there is not a clear definition for all the variables. For example in CD4Ini, the model with two variables includes the variable *Cul5-2*, which is not part of the model with six variables.

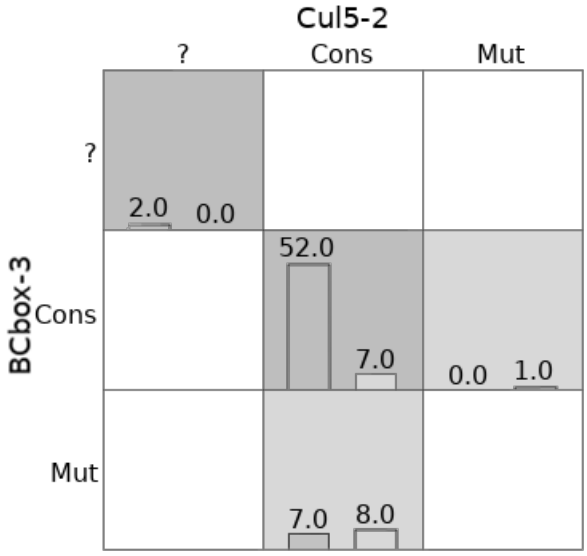


Figure 4.1 Example of the resulting cells considering the model size two learned by MDR on the CD4Ini output. The cells identified by the symbols "?" identify the sequences without values for the involved variables.

Table 4.7 Variables included in the selected MDR-models, shown with a value 1 per variable and output.

Output	**APOBEC-2**	**APOBEC-3**	**APOBEC-4**	**APOBEC-5**	**APOBEC-6**	**APOBEC-7**	**APOBEC-8**	**CBFb-1**	**CBFb-2**	**NLIS**	**Cul5-1**	**Cul5-2**	**Cul5-3**	**BCbox-1**	**BCbox-2**	**BCbox-3**
CD4Ini			1			1			1			1	1		1	1
CD4Hist	1	1	1			1			1				1		1	1
VLIni	1		1			1				1			1	1	1	
VLHist	1		1			1	1		1	1	1				1	1

4.1.3 C4.5 algorithm process

C4.5 decision tree induction was done using J48, an implementation in Weka. The experimentation was done using Confidence Level (CF) values in the range [0.1-0.51], subtree raising, a minimum number of objects per branch of two and 10-CV. The classification accuracy results per output were in the following ranges: [76.62%-80.52%] for CD4Ini, [72.73%-79.22%] for CD4Hist, [57.14%-62.34%] for VLIni and [53.25%-57.14%] for VLHist. These values were similar to those achieved by the baseline MLP, although some trees gave preference to the most represented class. Table 4.8 shows some details on the results from the inducted decision trees per output. The results show the number of classifying leaves and the tree sizes defined by the number of non-leaves nodes plus the classifying leaves (i.e. the total number of nodes).

Table 4.8 Classification performance and details from the induced trees with C4.5 per output: a) using CD4Ini; b) using CD4Hist; c) using VLIni; d) using VLHist.

a)

Output	CF Range	Accuracy %	Leaves	Tree size
CD4Ini	0.01-0.02	79.22	1	1
	0.03-0.09	77.92	1	1
	0.1-0.13	77.92	3	5
	0.14-0.14	77.92	4	7
	0.15-0.24	79.22	4	7
	0.25-0.25	79.22	5	9
	0.26-0.5	80.52	5	9
	0.51-0.51	76.62	11	21

b)

Output	CF Range	Accuracy %	Leaves	Tree size
CD4Hist	0.01-0.15	79.22	1	1
	0.16-0.25	77.92	1	1
	0.26-0.31	76.62	1	1
	0.32-0.36	75.32	1	1
	0.37-0.38	74.03	1	1
	0.39-0.47	74.03	5	9
	0.48-0.5	72.73	5	9
	0.51-0.51	72.73	11	21

c)

Output	CF Range	Accuracy %	Leaves	Tree size
VLIni	0.01-0.04	61.04	1	1
	0.05-0.06	58.44	1	1
	0.07-0.14	55.84	1	1
	0.15-0.17	57.14	1	1
	0.18-0.21	57.14	6	11
	0.22-0.24	58.44	6	11
	0.25-0.25	58.44	6	11
	0.26-0.27	59.74	7	13
	0.28-0.5	61.04	7	13
	0.51-0.51	62.34	8	15

d)

Output	CF Range	Accuracy %	Leaves	Tree size
VLHist	0.01-0.15	57.14	2	3
	0.16-0.24	58.44	2	3
	0.25-0.25	57.14	2	3
	0.26-0.28	55.84	2	3
	0.29-0.29	53.25	2	3
	0.3-0.3	53.25	5	9
	0.31-0.31	54.55	5	9
	0.32-0.32	55.84	5	9
	0.33-0.37	57.14	5	9
	0.38-0.4	57.14	10	19
	0.41-0.45	55.84	10	19
	0.46-0.47	54.55	10	19
	0.48-0.5	55.84	10	19
	0.51-0.51	54.55	10	19

When applying C4.5 on CD4Ini, ten inducted trees were uninformative (CF values [0.01-0.09]), as shown in Table 4.8a. Only four distinct trees were considered for posterior analysis with the CF values of 0.10, 0.14, 0.25 and 0.51, shown in Figure 4.2. The determination of the distinct trees was done considering the included variables, their depth in the tree, the tree

size and the number of leaves. All the selected trees included the variable *BCbox-3* while *APOBEC-4* or *APOBEC-7* and *Cul5-3* were present in three of them. The best classification performance was achieved with the CF values in the range [0.26-0.50], resulting in only one distinct inducted tree corresponding to that with a value of 0.25. Having the same tree with different classification is a normal behaviour in Weka since this is influenced by the CV random process and the imbalanced data, where some times the less represented class is not present in some of the partitions.

The C4.5 process for CD4Hist resulted in 38 uninformative decision trees for the CF range [0.01-0.38], as shown in Table 4.8b. Figure 4.3 shows the only two distinct informative inducted tress with the CF values 0.39 and 0.51. Both trees included the variables *APOBEC-2*, *APOBEC-7*, *BCbox-3* and *NLIS*. Considering the informative trees, the best classification performance was achieved with the CF values in the range [0.39-0.47].

Table 4.8c shows that VLIni seventeen inducted uninformative decision trees occurred in the CF range [0.01-0.17]. Four distinct informative inducted trees resulted in the CF values of 0.18, 0.25, 0.26 and 0.51. These four trees included the variables *APOBEC-2*, *BCbox-1*, *BCbox-2*, *CBFb-2* and *Cul5-3* and the one with the best classification had a CF value of 0.51.

None of the C4.5 inducted trees for VLHist was uninformative, as shown in Table 4.8d. This reflects that VLHist is the least imbalanced output, as shown in Table 1.2. However, only three of the 51 trees were distinct, with CF values of 0.01, 0.3 and 0.38 as shown in Figure 4.5. The trees with the best classification performance were those in the CF range [0.16-0.24].

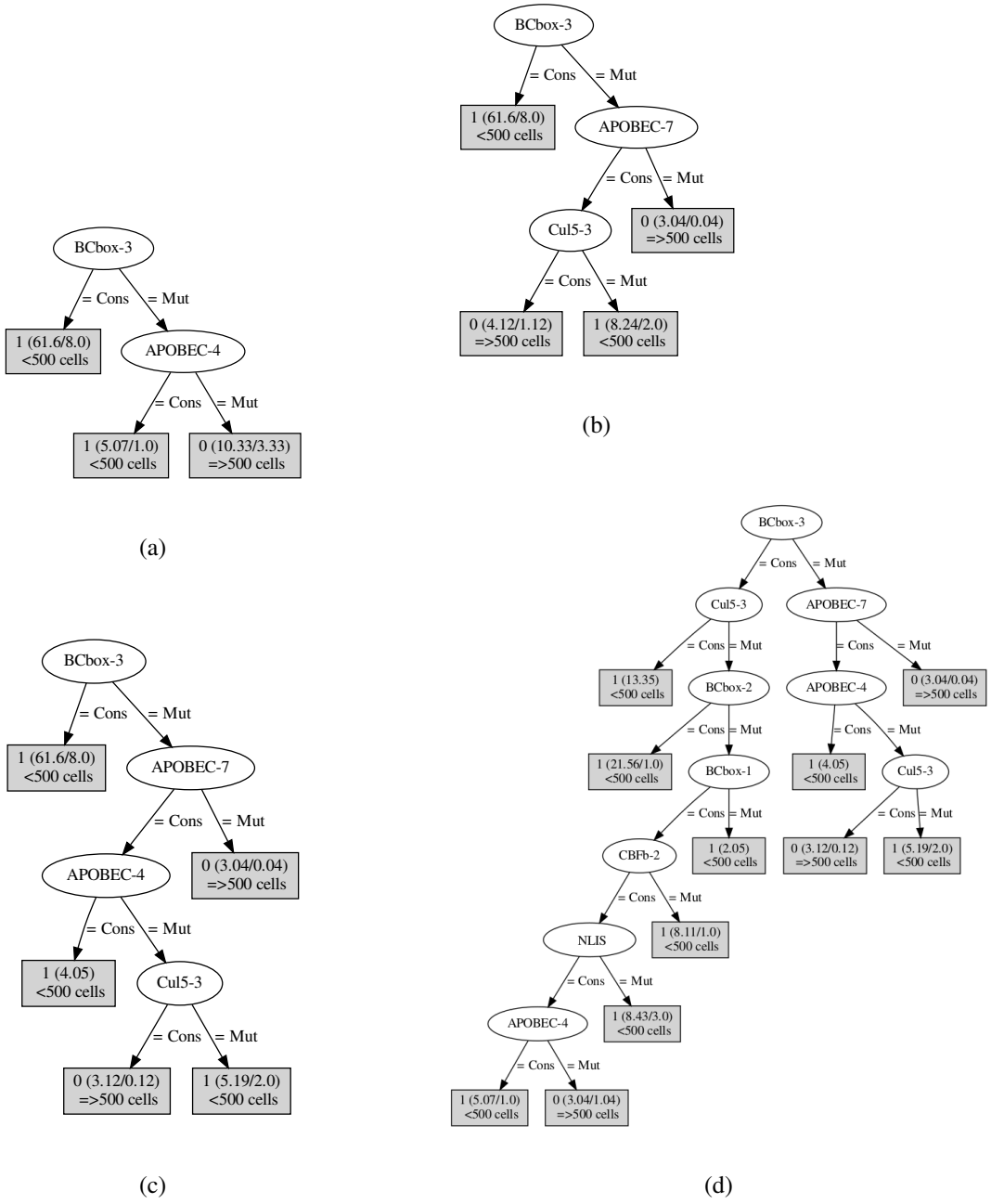
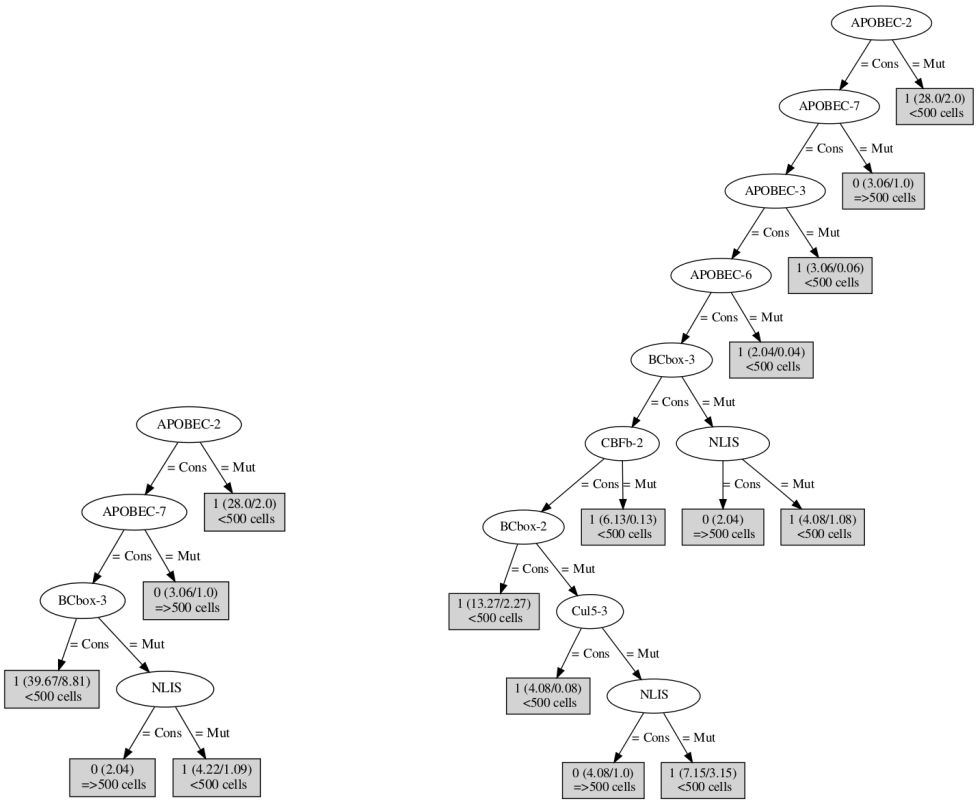


Figure 4.2 C4.5 distinct informative inducted trees on the output CD4Ini. a) CF value of 0.10. b) CF value of 0.14. c) CF value of 0.25. d) CF value of 0.51.



(a) (b)

Figure 4.3 C4.5 distinct informative inducted trees on the output CD4Hist. a) CF value of 0.39. b) CF value of 0.51.

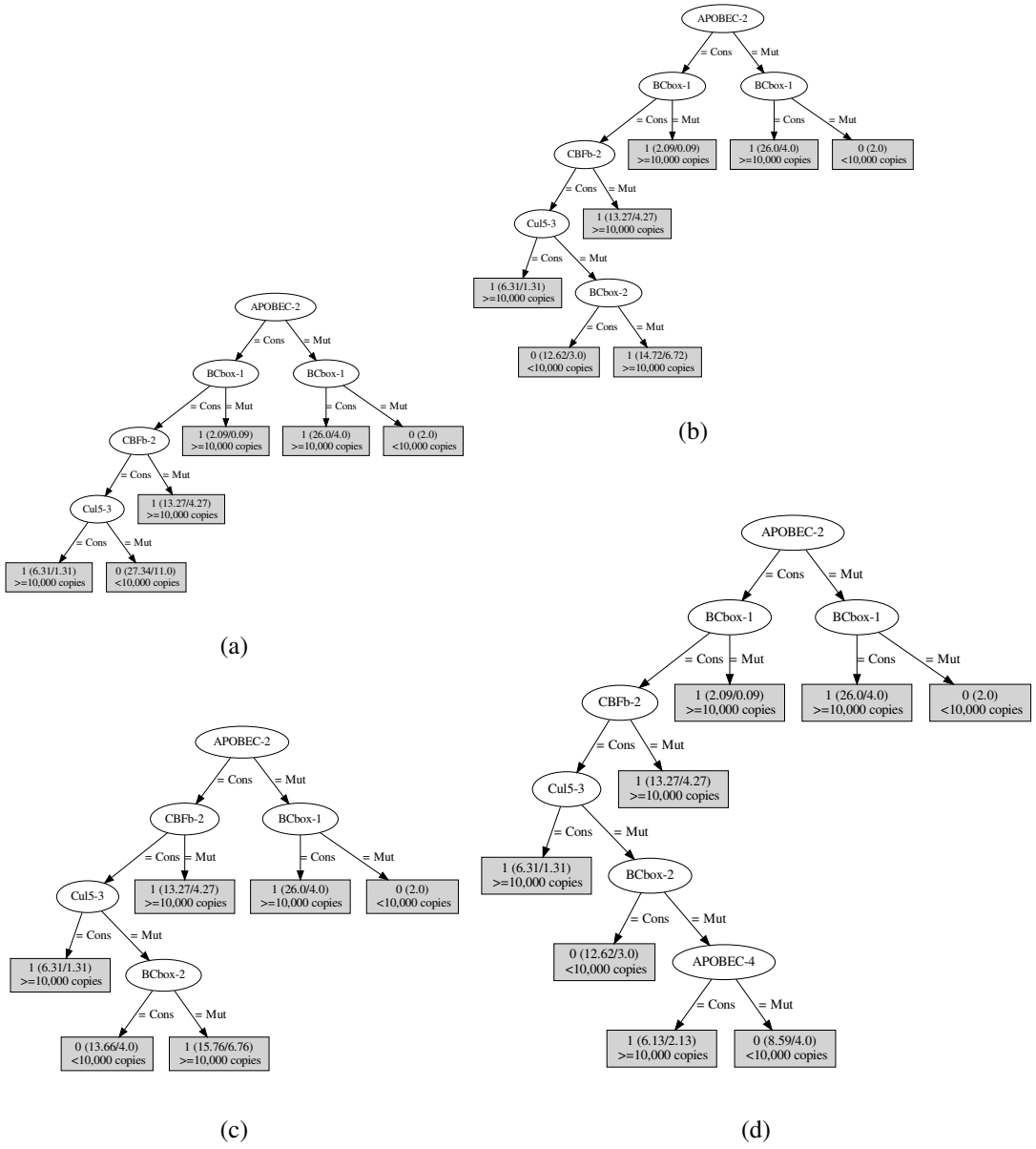


Figure 4.4 C4.5 distinct informative inducted trees on the output VLIni. a) CF value of 0.18. b) CF value of 0.25. c) CF value of 0.26. d) CF value of 0.51.

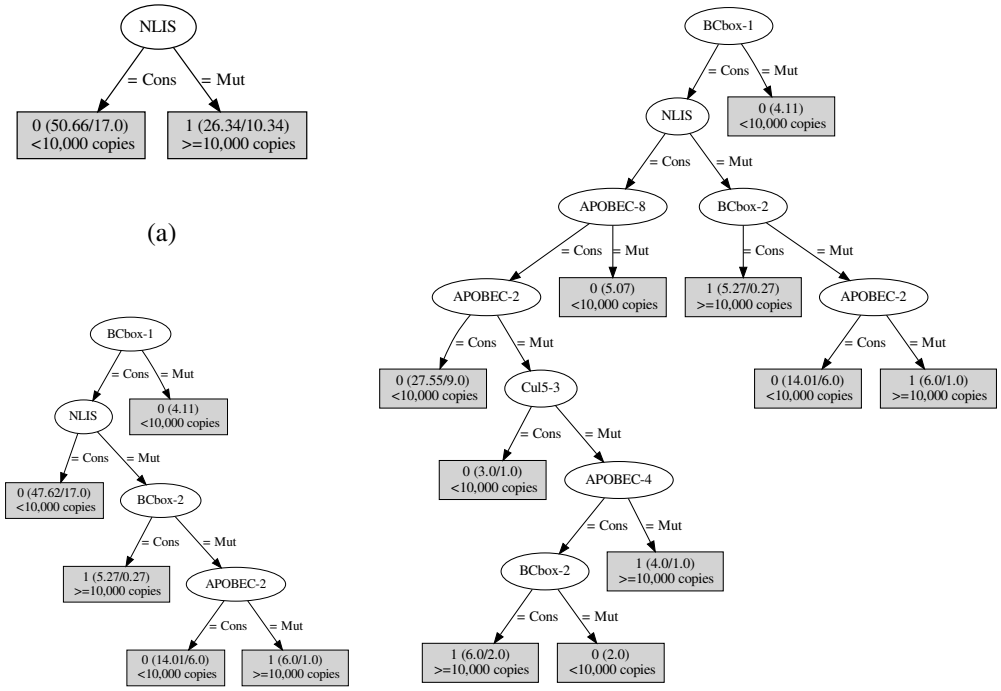


Figure 4.5 C4.5 distinct informative inducted trees on the output VLHist. a) CF value of 0.01. b) CF value of 0.30. c) CF value of 0.38.

The variables suggested as more relevant in by C4.5 are shown in Table 4.9. As with the selected Apriori-rules, CD4Ini and CD4Hist included the same number of different variables with eight each of them. In the case of VLIni, C4.5 included six variables while nine for VLHist. The selection of the variables was consistent, considering that those selected in smaller trees were included in the less pruned trees. Nevertheless, further analysis is required since seven variables that were selected in three of the four outputs (*APOBEC-2*, *APOBEC-4*, *CBFb-2*, *NLIS*, *Cul5-3*, *BCBox-1*, *BCBox-2* and *BCBox-3*). In this sense, evaluating the variables might allow the identification of those more related to each output.

Table 4.9 Variables included in the selected C4.5-decision trees branches, shown with a value 1 per variable and output.

Output	**APOBEC-2**	**APOBEC-3**	**APOBEC-4**	**APOBEC-5**	**APOBEC-6**	**APOBEC-7**	**APOBEC-8**	**CBFb-1**	**CBFb-2**	**NLIS**	**Cul5-1**	**Cul5-2**	**Cul5-3**	**BCbox-1**	**BCbox-2**	**BCbox-3**
CD4Ini			1			1			1	1			1	1	1	1
CD4Hist	1	1			1	1			1	1			1			1
VLIni	1		1						1				1	1	1	
VLHist	1		1				1			1			1	1	1	

4.1.4 Evaluating combinations with MLP

Once the Apriori-rules, MDR-models and C4.5-trees were generated, the next step involved evaluating all their specific combinations of variables in the MLP training process, as shown in 3.16b. For this, the previously mentioned configuration was used considering the number of variables in the combination as the value for *iv* (i.e. input variables for the training process, see Table 4.1). A specific combination is the description of the specific variables that are part of a specific rule, a specific model or a specific tree-branch. For example, the first Apriori-rule selected on the CD4Ini output yields the specific combination (*APOBEC-2*, *APOBEC-4*). Table 4.10 shows the resulting MLP evaluations on CD4Ini. These comprised the combinations from nine Apriori-rules, one MDR-model and ten C4.5 tree branches that improved the MLP-baseline performance (76.62%). The classification performance from these combinations was in the range [77.92%-85.71%]. Although none of the variables was present in all the improving combinations, some of them were selected consistently. These were *APOBEC-2*, *APOBEC-4*, *APOBEC-7*, *BCbox-3*. The best combination for the CD4Ini included *APOBEC-7*, *BCbox-3* and *Cul5-3*.

Using the output CD4Hist and the selected results from the other algorithms resulted in 19 Apriori-rules, three MDR-models and thirteen tree-branches combinations that improved the baseline performance (71.43%). Table 4.11 shows these results having a classification performance in the range [72.73%-81.82%]. All these combinations included the variable *APOBEC-2* while *APOBEC-3* was present in almost all of them. Eight combinations achieved the highest classification performance. These included the Apriori-rules numbers {1, 2, 3, 6}, one branch from the tree inducted using the CF-value of 0.39 and tree branches from the tree inducted using the CF-value of 0.51. All of these combinations included the variables *APOBEC-2* and *APOBEC-7* while *APOBEC-3* was present in six of them.

The fewer number of combination improving the MLP-baseline performance (64.94%) resulted when assessing the VLIni output (see Table 4.12). In this case, only one Apriori-rule and ten tree-branches combinations resulted in higher accuracy in the range [66.23%-71.43%]. None of the MDR-models improved the MLP-baseline performance. The variable *APOBEC-2* appeared in all the combinations improving the MLP-baseline performance, while *BCbox-1* appear in most of them. The combination *APOBEC-2*, *BCbox-1*, *BCbox-2*, *CBFb-2* and *Cul5-3* achieved the highest accuracy.

Using the VLHist output resulted in the lowest classification performance in the MLP assessment. The accuracy from twelve Apriori-rules improved the MLP-baseline performance (55.84%), as well as two from the MDR-models and eleven from the tree-branches. The

Table 4.10 MLP-evaluation on the selected rules, models and tree-branches from the Apriori, MDR and C4.5 algorithms using the CD4Ini output. Only the combinations resulting in a higher value than the MLP-baseline performance.

Algorithm	Rule/ Model/ CF-value	Variables	Accuracy %
MLP	-	All variables, except APOBEC-1 (baseline performance)	76.62
Apriori	1	APOBEC-2, APOBEC-4	79.22
	2	APOBEC-2, APOBEC-4, CBFb-1	79.22
	3	APOBEC-2, APOBEC-4, Cul5-1	79.22
	4	APOBEC-2, APOBEC-4, Cul5-2	77.92
	5	APOBEC-2, APOBEC-4, CBFb-1, Cul5-1	79.22
	8	APOBEC-2, APOBEC-4, CBFb-1, Cul5-1, Cul5-2	77.92
	9	APOBEC-2, APOBEC-3, APOBEC-4	79.22
	10	APOBEC-2, APOBEC-4, APOBEC-5	79.22
MDR	6	APOBEC-4, APOBEC-7, CBFb-2, Cul5-3, BCbox-2, BCbox-3	79.22
C4.5	0.1	BCbox-3, APOBEC-4	84.42
	0.14	BCbox-3, APOBEC-7, Cul5-3	85.71
	0.14	BCbox-3, APOBEC-7	77.92
	0.25	BCbox-3, APOBEC-7, APOBEC-4	79.22
	0.25	BCbox-3, APOBEC-7, APOBEC-4, Cul5-3	80.52
	0.25	BCbox-3, APOBEC-7	77.92
	0.51	BCbox-3, Cul5-3, BCbox-2, BCbox-1, CBFb-2	79.22
	0.51	BCbox-3, APOBEC-7, APOBEC-4	79.22
	0.51	BCbox-3, APOBEC-7, APOBEC-4, Cul5-3	80.52
	0.51	BCbox-3, APOBEC-7	77.92

performance was in the range [57.14%-67.53%]. The variables *APOBEC-2* and *NLIS* were present in most of the combinations of variables. The combination with the best accuracy was the one with *BCBox-1* and *NLIS*.

The number of variables combinations surpassing the value of baseline classification per output was 19 for CD4Ini, 35 for CD4Hist, eleven for VLIni and 25 for VLHist. In general, and considering the average performance of the combinations of the variables, the best ones were those suggested by C4.5, then the ones included in the MDR-models and in third place those present in the Apriori-rules. This was not the case for the VLIni output, where one Apriori-rule surpassed the baseline classification performance while none of the MDR-models did.

In the case of CD4Ini, the combinations identified by the rules had an average classification of 78.9%, a small improvement over the baseline performance of 76.64%. MDR only improved

Table 4.11 MLP-evaluation on the selected rules, models and tree-branches from the Apriori, MDR and C4.5 algorithms using the CD4Hist output. Only the combinations resulting in a higher value than the MLP-baseline performance.

Algorithm	Rule/ Model/ CF-value	Variables	Accuracy %
MLP		All variables (except APOBEC-1)	71.43
	1	APOBEC-2, APOBEC-3, APOBEC-7	81.82
	2	APOBEC-2, APOBEC-3, APOBEC-7, CBFb-1	81.82
	3	APOBEC-2, APOBEC-3, APOBEC-7, Cul5-1	81.82
	4	APOBEC-2, APOBEC-3, APOBEC-7, Cul5-2	77.92
	5	APOBEC-2, APOBEC-3, APOBEC-7, BCbox-1	80.52
	6	APOBEC-2, APOBEC-3, APOBEC-7, CBFb-1, Cul5-1	81.82
	7	APOBEC-2, APOBEC-3, APOBEC-7, CBFb-1, Cul5-2	80.52
	8	APOBEC-2, APOBEC-3, APOBEC-7, CBFb-1, BCbox-1	79.22
	9	APOBEC-2, APOBEC-3, APOBEC-7, Cul5-1, Cul5-2	77.92
Apriori	10	APOBEC-2, APOBEC-3, APOBEC-7, Cul5-1, BCbox-1	79.22
	133153	APOBEC-2, APOBEC-3, APOBEC-5, APOBEC-6, Cul5-3, BCbox-2	75.32
	133154	APOBEC-2, APOBEC-3, APOBEC-6, CBFb-1, Cul5-3, BCbox-2	77.92
	133155	APOBEC-2, APOBEC-3, APOBEC-6, Cul5-1, Cul5-3, BCbox-2	72.73
	133156	APOBEC-2, APOBEC-3, APOBEC-5, APOBEC-6, CBFb-1, Cul5-3, BCbox-2	74.03
	133157	APOBEC-2, APOBEC-3, APOBEC-5, APOBEC-6, Cul5-1, Cul5-3, BCbox-2	72.73
	133158	APOBEC-2, APOBEC-3, APOBEC-6, CBFb-1, Cul5-1, Cul5-3, BCbox-2	75.32
	133159	APOBEC-2, APOBEC-3, APOBEC-5, APOBEC-6, CBFb-1, Cul5-1, Cul5-3, BCbox-2	75.32
	133160	APOBEC-2, APOBEC-3, Cul5-3, BCbox-2	75.32
	133161	APOBEC-2, APOBEC-3, APOBEC-5, Cul5-3, BCbox-2	75.32
	1	APOBEC-2	79.22
MDR	2	APOBEC-2, APOBEC-3	79.22
	3	APOBEC-2, APOBEC-3, Cul5-3	79.22
	0.39	APOBEC-2, APOBEC-7, BCbox-3	76.62
	0.39	APOBEC-2, APOBEC-7, BCbox-3, NLIS	77.92
	0.39	APOBEC-2, APOBEC-7	81.82
	0.39	APOBEC-2	79.22
	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6, BCbox-3, CBFb-2, BCbox-2	75.32
	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6, BCbox-3, CBFb-2, BCbox-2, Cul5-3	77.92
C4.5	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6, BCbox-3, CBFb-2, BCbox-2, Cul5-3, NLIS	80.52
	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6, BCbox-3, CBFb-2	77.92
	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6, BCbox-3, NLIS	79.22
	0.51	APOBEC-2, APOBEC-7, APOBEC-3, APOBEC-6	81.82
	0.51	APOBEC-2, APOBEC-7, APOBEC-3	81.82
	0.51	APOBEC-2, APOBEC-7	81.82
	0.51	APOBEC-2	79.22

by using the model with six variables (79.22%). The best overall improvement for CD4Ini was through using the C4.5 decision trees-branches, having an average classification of 80.26%.

For the output CD4Hist, the baseline classification performance (71.43%) was clearly improved by the combinations suggested by three algorithms. The average classification performances were: 77.72% for the Apriori-rules, 79.22% for MDR and 79.32% for C4.5.

Table 4.12 MLP-evaluation on the selected rules, models and tree-branches from the Apriori, MDR and C4.5 algorithms using the VLIni output. Only the combinations resulting in a higher value than the MLP-baseline performance.

Algorithm	Rule/ Model/ CF-value	Variables	Accuracy %
MLP	-	All variables (except APOBEC-1)	64.94
Apriori	16422	APOBEC-2, APOBEC-3, APOBEC-5, NLIS, Cul5-3, BCbox-1, BCbox-2	66.23
MDR	-	-	-
C4.5	0.18	APOBEC-2, BCbox-1, CBFb-2, Cul5-3	67.53
	0.18	APOBEC-2, BCbox-1	66.23
	0.25	APOBEC-2, CBFb-2, Cul5-3, BCbox-2	67.53
	0.25	APOBEC-2, BCbox-1	66.23
	0.26	APOBEC-2, BCbox-1, CBFb-2, Cul5-3	67.53
	0.26	APOBEC-2, BCbox-1, CBFb-2, Cul5-3, BCbox-2	71.43
	0.26	APOBEC-2, BCbox-1	66.23
	0.51	APOBEC-2, BCbox-1, CBFb-2, Cul5-3	67.53
	0.51	APOBEC-2, BCbox-1, CBFb-2, Cul5-3, BCbox-2	71.43
	0.51	APOBEC-2, BCbox-1	66.23

The improvement was lower in the VLIni output, which had a baseline performance of 64.94%. In this case, only one Apriori-rule surpassed the baseline performance with 66.23%. The combinations from the C4.5 tree-branches averaged 67.79%. As mentioned before, none of the MDR-models suggested any combination capable of improving the baseline classification.

Similarly than with CD4Ini and CD4Hist, the MLP evaluation on VLHist showed an important improvement over the baseline classification of 55.84%. In this case, the combinations from the Apriori-rules averaged a classification performance of 59.85%, those from the MDR-models averaged 63.64% while those from the C4.5 tree-branches had in averaged 63.4%.

A summary on the variables included in the combinations that improved the baselines classification performances is shown in Table 4.14. Although some variables appear more consistently, there is not a clear indication of those that are more relevant. These results show that:

- for CD4Ini, the most constantly selected variables were *APOBEC-4* in 74% of the combinations, *BCBox-3* in 58% and *APOBEC-7* in 47%;
- for CD4Hist, the three most consistently selected variables were *APOBEC-2* on 100% of the times, *APOBEC-3* in 80% and *APOBEC-7* in 60%;

Table 4.13 MLP-evaluation on the selected rules, models and tree-branches from the Apriori, MDR and C4.5 algorithms using the VLHist output. Only the combinations resulting in a higher value than the MLP-baseline performance.

Algorithm	Rule/ Model/ CF-value	Variables	Accuracy %
MLP	-	All variables (except APOBEC-1)	55.84
Apriori	1	APOBEC-2, APOBEC-7, CBFb-2, NLIS, BCbox-2	58.44
	2	APOBEC-2, APOBEC-7, NLIS, Cul5-3, BCbox-2	61.04
	3	APOBEC-2, CBFb-1, CBFb-2, NLIS, BCbox-2	57.14
	4	APOBEC-2, CBFb-1, NLIS, Cul5-3, BCbox-2	59.74
	9	APOBEC-2, APOBEC-5, APOBEC-7, CBFb-2, NLIS, BCbox-2	57.14
	10	APOBEC-2, APOBEC-5, APOBEC-7, NLIS, Cul5-3, BCbox-2	59.74
	10916	APOBEC-2, APOBEC-4, APOBEC-8, BCbox-1	61.04
	10939	APOBEC-2, APOBEC-4, APOBEC-5, APOBEC-8, BCbox-1	61.04
	10940	APOBEC-2, APOBEC-4, APOBEC-8, CBFb-1, BCbox-1	61.04
	10941	APOBEC-2, APOBEC-4, APOBEC-8, Cul5-1, BCbox-1	61.04
	10942	APOBEC-2, APOBEC-4, APOBEC-8, Cul5-2, BCbox-1	61.04
	11061	APOBEC-2, APOBEC-3, APOBEC-4, APOBEC-8, CBFb-1, BCbox-1	59.74
MDR	1	NLIS	63.64
	2	NLIS, Cul5-1	63.64
C4.5	0.3	BCbox-1, NLIS	67.53
	0.3	BCbox-1, NLIS, BCbox-2	66.23
	0.3	BCbox-1, NLIS, BCbox-2, APOBEC-2	66.23
	0.3	BCbox-1	57.14
	0.38	BCbox-1, NLIS, APOBEC-8, APOBEC-2	63.64
	0.38	BCbox-1, NLIS, APOBEC-8, APOBEC-2, Cul5-3	62.34
	0.38	BCbox-1, NLIS, APOBEC-8, APOBEC-2, Cul5-3, APOBEC-4, BCbox-2	59.74
	0.38	BCbox-1, NLIS, APOBEC-8	64.94
	0.38	BCbox-1, NLIS, BCbox-2	66.23
	0.38	BCbox-1, NLIS, BCbox-2, APOBEC-2	66.23
	0.38	BCbox-1	57.14

- for VLIni, the most selected variable was *APOBEC-2* in 100% of the combinations, *BCBox-1* in 91% and *Cul5-3* in 64%;
- for VLHist, the three variables *APOBEC-2*, *NLIS* and *BCBox-1* were included in 68% of the combinations.

Table 4.14 Percentages considering the variables inclusion in combinations that improved the MLP classification performance per output.

Output	APOBEC-2	APOBEC-3	APOBEC-4	APOBEC-5	APOBEC-6	APOBEC-7	APOBEC-8	CBFb-1	CBFb-2	NLIS	Cul5-1	Cul5-2	Cul5-3	BCbox-1	BCbox-2	BCbox-3
CD4Ini	42%	5%	74%	5%	0%	47%	0%	16%	11%	0%	16%	11%	26%	5%	11%	58%
CD4Hist	100%	80%	0%	14%	37%	60%	0%	23%	11%	9%	23%	9%	34%	9%	34%	20%
VLIni	100%	9%	0%	9%	0%	0%	0%	0%	55%	9%	0%	0%	64%	91%	36%	0%
VLHist	68%	4%	28%	12%	0%	16%	40%	16%	12%	68%	8%	4%	20%	68%	44%	0%

4.1.5 Evaluating variable relevance using a points-based system

After testing the selected Apriori-rules, MDR-models and C4.5 tree-branches with MLP, the next step was analysing the variable relevance per output. To make the variables influence on each class clearer, the analysis included the assessment according to the class related to each rule, model or tree branch. In the case of the rules, the head indicates the class. We followed two approaches for the MDR-models and C4.5 tree-branches. For the MDR-models, only those combinations resulting in cells identifying at least five sequences were considered, associating the cells to the class with more representation in them. In this way, the cells identifying very few examples are removed from further analysis. For example, the two variable MDR-model on the CD4Ini output had the cell determined by *Cul5-2*$^{\text{Cons}}$ and *BCbox-3*$^{\text{Mut}}$ with eight examples of Class=0 and seven of Class=1. The learned model for this example is shown in Figure 4.1. Therefore, this combination was considered as from Class=0. Similarly, the separation by class with the C4.5 tree-branches considered only the leaves having an estimation with over five sequences identified after removing the estimated errors. Here, each classifying leaf defined the class.

Our methodology calculates the importance of the variables by using the points defined in for the methodology (see section 3.2), which are:

1. Points assigned for the Apriori algorithm:
   (a) Two points for each time a variable is present in one of the selected rules.
   (b) The proportion of 20 points considering variables in selected rules that improve the MLP-baseline classification performance in relation to all the selected rules.
2. Points assigned for the C4.5 algorithm:

(a) Six points are added to each variable considering the tree-branches in which it is included.

(b) Considering the variable included in the trees, three additional proportions are considered:

i. The proportion of ten points considering in how many of the distinct trees each variable was included.

ii. The proportion of ten points considering the rank that each variable holds in the distinct trees. This value is calculated using Equation 3.17. The proportion depends on the average level at which the variable is found.

iii. A proportion of 30 points by considering how many of the classification-branches that improve the baseline classification include the variable.

3. Points assigned for the MDR algorithm:

(a) Eight points to each variable present in each of the cells from the best models. This is similar to finding the variable in a decision tree branch.

(b) The proportion of 20 points considering in how many of the selected models with combinations improving the baseline performance appears when considering all the possible combinations on the selected models.

Table 4.15 shows the summaries after analysing the results from the machine learning algorithms for the initial and historical CD4+ cell counts. The columns headers ≥ 500 cells$/\mu L$ indicate results for the Class "0", while those with < 500 cells$/\mu L$ for "1". Similarly, Table 4.16 shows the resulting summaries for the initial and historical viral loads. These labels with $<$ 10K cp/mL indicate results for the Class "0", while those with $\geq$ 10K cp/mL for "1". Both of these tables summarise:

- The number of times that a variable was included in the Apriori-rules per class.
- The number of times that a variable was included in the MDR-models per class.
- In the case of C4.5, the columns describe:
  - The number of times a variable was included in the distinct inducted decision trees per class.
  - The fraction of the trees that included the variable.

  - The average rank position for every variable present in the trees.

- For the MLP results, the columns describe the fraction of combinations including the variable that improved the baseline performance considering:
  - Apriori-rules.
  - MDR-models.
  - the branches of the C4.5-decision trees.

- The columns labelled as "Weighted contribution" show the calculation of points per class and variable.

The methodology defined the most relevant variables per output by using the values in the columns "Weighted contribution". These variables are used in the next stage for determining the specific combinations to be delivered by as hypotheses.

Table 4.15 Summary per variable on the outputs related to the CD4+ T cells count. The table cells with fill colour indicate the variables with the highest scores.

		Apriori		MDR		C4.5								
		Occurence in rules		Occurence in models		Occurrence in branches		Occurence in different trees	Averange rank in trees	Occurrence in MLP			Weighted contribution	
	Variable	≥500 cells/μL	<500 cells/μL	≥500 cells/μL	<500 cells/μL	≥500 cells/μL	<500 cells/μL			Apriori	MDR	C4.5	≥500 cells/μL	<500 cells/μL
Initial CD4+ T lymphocytes (CD4ini)	APOBEC-2	-	10	-	-	-	-	0/4	-	8/8	0/3	0/10	20	40
	APOBEC-3	8	1	-	-	-	-	0/4	-	1/8	0/3	0/10	18.5	4.5
	APOBEC-4	-	10	-	6	1	-	3/4	4	8/8	1/3	5/10	60.19	122.19
	APOBEC-5	2	1	-	-	-	-	0/4	-	1/8	0/3	0/10	6.5	4.5
	APOBEC-6	1	-	-	-	-	-	0/4	-	0/8	0/3	0/10	2	-
	APOBEC-7	-	-	-	6	-	1	3/4	4	0/8	1/3	8/10	43.19	97.12
	APOBEC-8	3	-	-	-	-	-	0/4	-	0/8	0/3	0/10	6	-
	CBFb-1	1	4	-	-	-	-	0/4	-	3/8	0/3	0/10	9.5	15.5
	CBFb-2	-	-	-	6	-	2	1/4	5	0/8	1/3	1/10	15.51	75.44
	NLIS	-	-	-	-	-	1	1/4	6	0/8	0/3	0/10	4.17	10.16
	Cul5-1	1	4	-	-	-	-	0/4	-	3/8	0/3	0/10	9.5	15.5
	Cul5-2	1	4	1	1	-	-	0/4	-	2/8	0/3	0/10	15	21
	Cul5-3	-	-	-	6	-	5	3/4	3	0/8	1/3	4/10	32.84	110.81
	BCbox-1	1	-	-	-	-	2	1/4	4	0/8	0/3	1/10	12.5	22.5
	BCbox-2	-	-	-	6	-	3	1/4	3	0/8	1/3	1/10	18.84	84.81
	BCbox-3	10	-	2	8	1	8	4/4	1	0/8	1/3	10/10	98.62	168.75
Historical CD4+ T lymphocytes (CD4Hist)	APOBEC-2	10	10	-	8	-	5	2/2	1	19/19	3/3	13/13	138	234
	APOBEC-3	10	10	-	6	-	2	1/2	3	19/19	2/3	7/13	30	40
	APOBEC-4	-	-	-	-	-	-	0/2	-	0/19	0/3	0/13	4.17	4.17
	APOBEC-5	5	-	-	-	-	-	0/2	-	5/19	0/3	0/13	24	28
	APOBEC-6	8	-	-	-	-	2	1/2	4	7/19	0/3	6/13	4	4
	APOBEC-7	-	10	-	-	-	3	2/2	2	10/19	0/3	11/13	16	84
	APOBEC-8	-	-	-	-	-	-	0/2	-	0/19	0/3	0/13	-	-
	CBFb-1	4	4	-	-	-	-	0/2	-	8/19	0/3	0/13	-	4
	CBFb-2	-	-	-	-	-	2	1/2	6	0/19	0/3	4/13	60.34	92.31
	NLIS	-	-	-	-	-	-	2/2	6.5	0/19	0/3	3/13	64	124
	Cul5-1	4	4	-	-	-	-	0/2	-	8/19	0/3	0/13	4	4
	Cul5-2	-	3	-	-	-	-	0/2	-	3/19	0/3	0/13	4	2
	Cul5-3	10	-	-	4	-	-	1/2	8	9/19	1/3	2/13	98.69	66.62
	BCbox-1	-	3	-	-	-	-	0/2	-	3/19	0/3	0/13	111.31	147.25
	BCbox-2	10		-	-	-	1	1/2	7	9/19	0/3	3/13	103.5	157.5
	BCbox-3	-	-	-	-	-	3	2/2	4	0/19	0/3	7/13	-	20

Table 4.16 Summary per variable on the outputs related to the viral load. The table cells with fill colour indicate the variables with the highest scores.

		Apriori		MDR		C4.5								
		Occurence in rules		Occurence in models		Occurrence in branches		Occurence in different trees	Averange rank in trees	Occurrence in MLP			Weighted contribution	
	Variable	<10K cp/mL	≥10K cp/mL	<10K cp/mL	≥10K cp/mL	<10K cp/mL	≥10K cp/mL			Apriori	MDR	C4.5	<10K cp/mL	≥10K cp/mL
Initial viral load (VLIni)	APOBEC-2	10	-	3	13	4	10	4/4	1	1/1	0/3	10/10	110	204
	APOBEC-3	5	10	-	-	-	-	0/4	-	1/1	0/3	0/10	82	142
	APOBEC-4	-	-	-	-	-	-	1/4	6	0/1	0/3	0/10	-	-
	APOBEC-5	2	4	-	-	-	-	0/4	-	1/1	0/3	0/10	15.27	5.26
	APOBEC-6	2	2	-	-	-	-	0/4	-	0/1	0/3	0/10	48.47	44.47
	APOBEC-7	-	10	2	8	-	-	0/4	-	0/1	0/3	0/10	54.69	92.62
	APOBEC-8	-	-	-	-	-	-	0/4	-	0/1	0/3	0/10	-	-
	CBFb-1	-	2	-	-	-	-	0/4	-	0/1	0/3	0/10	16.42	16.42
	CBFb-2	-	10	-	-	4	6	4/4	2.75	0/1	0/3	6/10	17.98	29.98
	NLIS	10	-	3	13	-	-	0/4	-	1/1	0/3	0/10	20.67	20.67
	Cul5-1	2	2	-	-	-	-	0/4	-	0/1	0/3	0/10	16.42	16.42
	Cul5-2	2	1	-	-	-	-	0/4	-	0/1	0/3	0/10	3.16	9.16
	Cul5-3	10	-	-	-	4	2	4/4	3.75	1/1	0/3	6/10	47	59
	BCbox-1	10	1	1	4	3	8	4/4	2.75	1/1	0/3	9/10	3.16	9.16
	BCbox-2	10	-	3	13	3	2	3/4	4.67	1/1	0/3	3/10	43.88	29.89
	BCbox-3	-	10	-	-	-	-	0/4	-	0/1	0/3	0/10	32.41	50.41
Historical viral load (VLHist)	APOBEC-2	10	10	-	-	3	-	2/2	4	12/12	0/2	5/11	86.62	68.62
	APOBEC-3	4	5	-	-	-	-	0/2	-	1/12	0/2	0/11	9.66	11.66
	APOBEC-4	-	10	-	-	-	-	1/2	6	6/12	0/2	1/11	19.41	39.38
	APOBEC-5	2	2	-	-	-	-	0/2	-	3/12	0/2	0/11	9	9
	APOBEC-6	-	-	-	-	-	-	0/2	-	0/12	0/2	0/11	-	-
	APOBEC-7	6	-	-	-	-	-	0/2	-	4/12	0/2	0/11	18.67	6.67
	APOBEC-8	-	10	-	-	2	-	1/2	3	6/12	0/2	4/11	44.56	52.56
	CBFb-1	4	2	-	-	-	-	0/2	-	4/12	0/2	0/11	14.67	10.67
	CBFb-2	5	-	-	-	-	-	0/2	-	3/12	0/2	0/11	15	5
	NLIS	10	-	2	2	5	-	2/2	2	6/12	2/2	9/11	138.88	88.88
	Cul5-1	-	2	1	1	-	-	0/2	-	1/12	1/2	0/11	19.66	23.67
	Cul5-2	-	2	-	-	-	-	0/2	-	1/12	0/2	0/11	1.67	5.67
	Cul5-3	5	-	-	-	-	-	1/2	5	3/12	0/2	2/11	28.78	18.78
	BCbox-1	-	10	-	-	5	-	2/2	1	6/12	0/2	11/11	90	80
	BCbox-2	10	-	-	-	2	-	2/2	5	6/12	0/2	5/11	69	36.97
	BCbox-3	-	-	-	-	-	-	0/2	-	0/12	0/2	0/11	-	-

The selection of the three most relevant variables per output was made considering their influence over the Class 1 (i.e. the patients more affected). This was decided since these sequences are the ones more represented in the outputs (except VLHist), while also being the most interesting cases. On the other hand, the accumulated points showed that there were sets of three variables having the highest values (see Figures 4.6 and 4.7). For this reason, it was decided to set the number of variables to at most three. This also helps in being more consistent with the statistical parsimony, considering smaller models easier for explaining [144, 170]. Furthermore, as mentioned by Lustgarten et al., in biomarker discovery it is preferred to have fewer variables to describe the data. This is because the validation on those markers might be time-consuming and expensive [97].

Figures 4.6b, 4.6d, 4.7b and 4.7d show for each output the variable assessment on Class 1 based on Tables 4.15 and 4.16. On the other hand, Figures 4.6a, 4.6c, 4.7a and 4.7c show the relevance when considering Class 0. The same variables appeared consistently for both classes in most of the outputs, which suggests that these variables are the most relevant. The

summaries include the points determined for each algorithm considering their resulting rules, models or trees. These values include the points assigned from the MLP evaluations.

Considering the CD4Ini output, the variables more relevant for the sequences with $<$500 cell/μL (Class 1) were *BCbox-3*, *APOBEC-4* and *Cul5-3*, as shown in Figure 4.6b. *BCbox-3* and *Cul5-3* did not appear in any of the top ten Apriori-rules for Class 1. Nevertheless, they were highly important in the MDR-models and C4.5 trees. On the other hand, *APOBEC-4* appeared more consistently. These variables were also more relevant for the sequences with $\geq$500 cell/μL (Class 0) except *Cul5-3* that was less important than *APOBEC-7* as shown in Figure 4.6a.

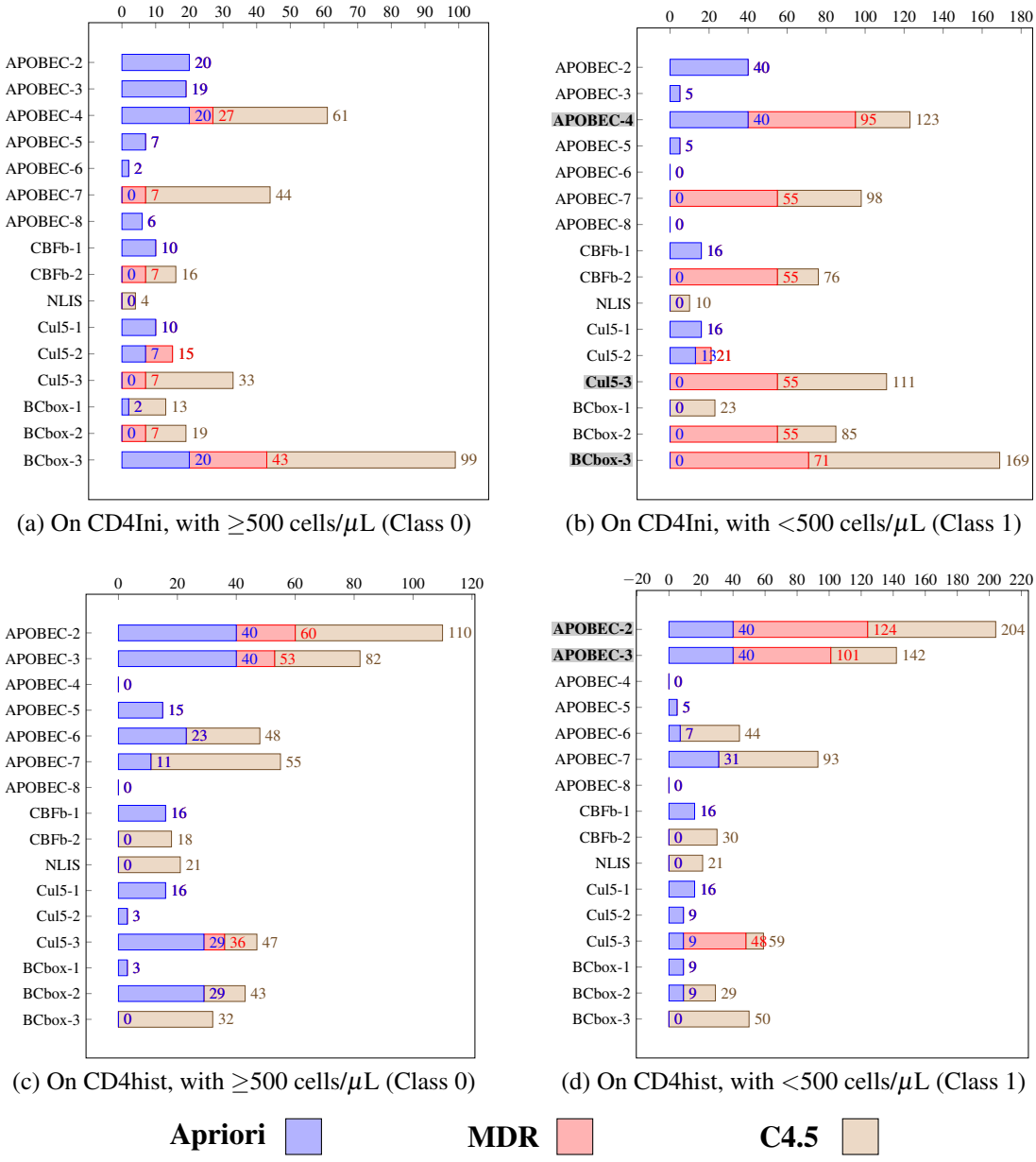
With CD4Hist, the most relevant variables for both classes were *APOBEC-2* and *APOBEC-3*. As can been seen in Figures 4.6c and 4.6d, the importance of both variables was suggested by the three algorithms. In the case of CD4Hist, a third variable (*APOBEC-7*) was discarded since its accumulation of points (93) was far from the one with the higher value (204). Furthermore, this variable was not included in the best MDR-models.

The three most relevant variables with VLIni were *APOBEC-2*, *BCbox-2* and *BCbox-1*. These variables were relevant for both classes while being consistently selected by the three algorithms as shown in Figures 4.7a and 4.7b. Additional variables showed an influence on the classes, for example, *NLIS* reflected involvement for the sequences with $\geq$10,000 virus copies per millilitre (Class 1), while *Cul5-3* appear as important for the sequences with less $<$10,000 virus copies per millilitre (Class 0). The analysis with the VLHist output suggested the variables *NLIS*, *BCbox-1* and *APOBEC-2* as the most relevant. Only *NLIS* achieved support from the three algorithms, while *BCbox-1* and *APOBEC-2* were not present in the MDR-models. Nevertheless, these three variables were the most relevant for both classes as shown in Figures 4.7c and 4.7d.

As mentioned before, the most relevant variables were identified considering the total points achieved per output. Table 4.17 shows the selected variables and their relevance rank. The results suggest great importance for *APOBEC-2* since it appears as relevant for all the outputs except for the initial CD4+ T cell count. Similarly, *BCbox-1* appears as relevant considering the initial and historical viral loads.

4.1.6 Generation of decision trees using ID3 for hypotheses testing

After selecting the most relevant variables, the next step in the methodology involves their use as input in the algorithm ID3. As mentioned before, ID3 was preferred over C4.5 since we are interested in detecting the relevant variables and values combinations without using any



(a) On CD4Ini, with ≥500 cells/μL (Class 0)

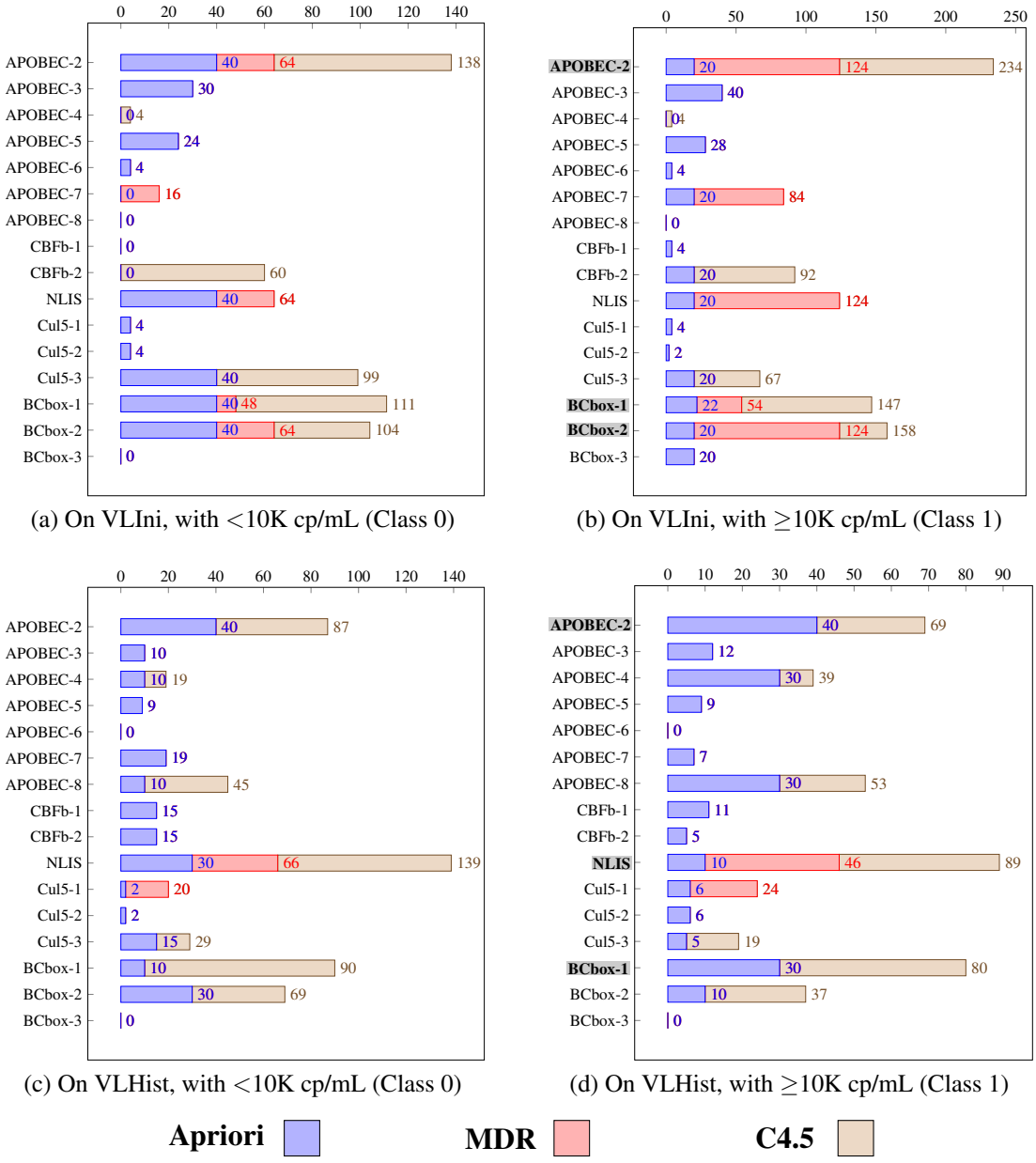
(b) On CD4Ini, with <500 cells/μL (Class 1)

(c) On CD4hist, with ≥500 cells/μL (Class 0)

(d) On CD4hist, with <500 cells/μL (Class 1)

Figure 4.6 Variable assessment by output and algorithm considering the CD4+ T cells initial and historical counts. Each bar shows the total number of points achieved per variable while each colour shows the source: Apriori-rules (blue), MDR-models (red) or C4.5 trees (yellow). The name of the selected variables considering the class "1" are shown in highlighted bold font.

pruning strategy. On the other hand, ID3 was preferred over MDR since their results are similar and of easier interpretation. In this way, the specific variables-values combinations to be used



(a) On VLIni, with <10K cp/mL (Class 0)

(b) On VLIni, with ≥10K cp/mL (Class 1)

(c) On VLHist, with <10K cp/mL (Class 0)

(d) On VLHist, with ≥10K cp/mL (Class 1)

Figure 4.7 Variable assessment by output and algorithm considering the initial and historical viral loads. Each bar shows the total number of points achieved per variable while each colour shows the source: Apriori-rules (blue), MDR-models (red) or C4.5 trees (yellow). The name of the selected variables considering the class "1" are shown in highlighted bold font.

as hypotheses are defined. By using this approach there is no need for exploring all the possible combinations but only those identified by ID3. In this process, all but two of the sequences

Table 4.17 Variables suggested as more relevant by the methodology. The number indicates the importance rank that the variable represented in each output.

Output	APOBEC-2	APOBEC-3	APOBEC-4	BCbox-1	Bcbox-2	Bcbox-3	Cul5-3	NLIS
CD4Ini			2^{nd}			1^{st}	3^{rd}	
CD4Hist	1^{st}	2^{nd}						
VLIni	1^{st}			2^{nd}	3^{rd}			
VLHist	3^{rd}			2^{nd}				1^{st}

were used considering the variables previously selected for each output. Two sequences were excluded since ID3 can not work with empty values. Figure 4.8 shows the resulting trees for each output.

There are seven distinct suggested hypotheses for CD4Ini, one per each branch as shown in Fig. 4.8a. These hypotheses involve the variables *BCbox-3*, *APOBEC-4* and *Cul5-3*. In the case of CD4Hist, only four hypotheses are suggested since only two variables were selected as relevant: *APOBEC-2* and *APOBEC-3*. With VLIni, six different hypotheses were suggested on the variables *APOBEC-2*, *BCbox-1* and *BCbox-2*. Similarly, for VLHist only six hypotheses are suggested when using the variables *NLIS*, *BCbox-1* and *APOBEC-2* as input. The total number of suggested hypotheses was 23. Although the Apriori, C4.5 and MDR had already generated combinations that could be used as hypotheses, it is desirable to avoid evaluating all of them since this would require a further correction on the hypotheses as associated. Furthermore, applying the test on only those combinations delivered from the completed methodology has the advantage of using the collected knowledge on the more relevant variables.

4.1.7 Statistical analysis on the resulting combinations from the methodology

Next are presented the results on testing the suggested combinations from the methodology. The estimation of the possible number of combinations was made using the process explained in section 1.5. Considering the HIV-vif-dataset, the total number of hypotheses to test after removing the variable *APOBEC-1* is 43,046,720 for each output. In contrast, by employing our proposed methodology, the total number of resulting combinations for testing is only 23. These

Figure 4.8 ID3 inducted trees using the selected most relevant variables per output. a) The tree for CD4Ini; b) The tree for CD4Hist; c) The tree for VLIni; d) The tree for VLHist.

suggested hypotheses reflect the variables found as most relevant and their most informative combinations.

The test on the suggested hypotheses from the methodology was made by using Fisher's exact test. The process for the calculation using this traditional statistical test is explained in Annexe A. The significance level was $\alpha < 0.05$ using contingency tables after removing two sequences with missing data. Table 4.18 shows the results including at least one significant combination for each output. The table also shows the classification accuracy and error on using the combinations for the identification of sequences in the dataset.

The classification accuracy and error were calculated according to the suggested combination. In the table, the rows indicating "*absent*" describe the number of sequences that did not have the specific variables-values combinations. On the other hand, the rows labelled as "*present*" indicate the number of sequences that did have a specific combination. The accuracy

on combinations suggesting *protection* was calculated by summing the number of sequences in the cell defined by the column Class=0 and the row *present* plus the value in the cell Class=1 and row *absent* (i.e. matrix antidiagonal). Similarly, the classification error required the sum of the values in the cells Class=0 and the row *absent* plus the value in the cell Class=1 and row *present* (i.e. the main diagonal). Each of these sums was then divided by the total sequence number in the analysis. For the combinations suggesting *risk*, the sums were calculated in the inverse order: accuracy by the main diagonal and the error by the antidiagonal. Likewise, both values were divided by the total number of sequences on the analysis. In this sense, when considering protection, the summations reflect the number of sequences having the combination "*present*" related to the class with better health status (i.e. Class 0) or of those having the combination "*absent*" in relation to the class with worst status. The same applies when considering risk, the summations reflect the number of sequences having the combination "*present*" related to the class with worst health status (i.e. Class 1) or of those having the combination "*absent*" in relation to the class with better status.

All the combinations described next achieved a p-value lower than that of the significance threshold ($\alpha = 0.05$). Only two of the tested combinations, one from CD4Ini and one from CD4Hist, achieved significance levels lower than 0.01. Considering the classification performance, the lowest errors were achieved on CD4Ini (17.3%, 20%) while the worst appear on the rest of the outputs: CD4Hist (44%, 46.7%), VLIni (49.3%) and VLHist (37.3%).

The best combination for CD4Ini was *BCbox-3*$^{\text{Mut}}$, *APOBEC-4*$^{\text{Mut}}$ and *Cul5-3*$^{\text{Cons}}$ with a p-value of 0.0083. The absence of this combination (i.e. having the values *BCbox-3*$\neq 1$ or *APOBEC-4*$\neq 1$ or *Cul5-3*$\neq 0$) allowed identifying the 59 cases of Class 1 ($<$ 500 CD4+ T cells/μL). On the other hand, the presence of said combination (i.e. having the values *BCbox-3*=1 and *APOBEC-4*=1 and *Cul5-3*=0) suggests that the sequences can be considered as of Class 0 ($\geq$ 500 CD4+ T cells/μL). For this reason, the combination can be considered protective. The second-best hypothesis with CD4Ini includes *BCbox-3*$^{\text{Mut}}$, *APOBEC-4*$^{\text{Mut}}$ and *Cul5-3*$^{\text{Mut}}$ with a p-value of 0.0338. This combination also suggests protection since its presence had a higher proportion of cases on Class 0 and three of the other class. On the contrary, its absence reflects almost all the sequences from Class 1.

The tests on CD4Hist had the lowest p-value from all the results and it involved the combination *APOBEC-2*$^{\text{Mut}}$ and *APOBEC-3*$^{\text{Cons}}$. Although this combination had a high classification error (46.7%), its presence identified a higher proportion of Class 1 sequences (i.e. 34) and only one of Class 0. This suggests a risk effect since most of the sequences of Class 0 are excluded when the combination is present. The second-best combination for CD4Hist, on the contrary,

Table 4.18 Combinations detected as associated after the statistical assessment ($\alpha < 0.05$) on the 23 suggested hypotheses. At least one association was found per output when the Fisher's exact test considering a significance level of $\alpha < 0.05$. The combinations marked as (^{p}) suggest protection, while the ones marked as (^{r}) suggest risk.

Output	Vif attribute combination		Contingency tables ≥500 cells/μL	Contingency tables <500 cells/μL	Classification Accuracy	Classification Error	p-value$^{\text{effect}}$
Initial CD4	BCbox-3$^{\text{Mut}}$, APOBEC-4$^{\text{Mut}}$, Cul5-3$^{\text{Cons}}$	absent	13	59	82.7%	17.3%	**0.0083^{p}**
		present	3	0	(62/75)	(13/75)	
	BCbox-3$^{\text{Mut}}$, APOBEC-4$^{\text{Mut}}$, Cul5-3$^{\text{Mut}}$	absent	12	56	80%	20.0%	**0.0338^{p}**
		present	4	3	(60/75)	(15/75)	
Historic CD4	APOBEC-2$^{\text{Mut}}$, APOBEC-3$^{\text{Cons}}$	absent	14	34	53.3%	46.7%	**0.0074^{r}**
		present	1	26	(40/75)	(35/75)	
	APOBEC-2$^{\text{Cons}}$, APOBEC-3$^{\text{Cons}}$	absent	2	29	56%	44.0%	**0.0182^{p}**
		present	13	31	(42/75)	(33/75)	
			<10,000 cp/mL	≥10,000 cp/mL			
Initial Viral Load (VL)	APOBEC-2$^{\text{Cons}}$, BCbox-1$^{\text{Cons}}$, BCbox-2$^{\text{Cons}}$	absent	15	40	68%	32.0%	**0.0318^{p}**
		present	11	9	(51/75)	(24/75)	
	APOBEC-2$^{\text{Mut}}$, BCbox-1$^{\text{Cons}}$, BCbox-2$^{\text{Mut}}$	absent	24	35	50.6%	49.3%	**0.0415^{r}**
		present	2	14	(38/75)	(37/75)	
Historic VL	NLIS$^{\text{Mut}}$, BCbox-1$^{\text{Cons}}$, APOBEC-2$^{\text{Mut}}$	absent	41	27	62.6%	37.3%	**0.0392^{r}**
		present	1	6	(47/75)	(28/75)	

suggests protection since the presence of *APOBEC-2*$^{\text{Cons}}$ and *APOBEC-3*$^{\text{Cons}}$ allows identifying on thirteen of the fifteen sequences from Class 0. Contrarily, its absence only identified two sequences of Class 0 while being related with 29 of the 60 of Class 1. This hypothesis had a p-value of 0.0182.

The lowest p-value with VLIni (0.0318) was achieved considering the combination *APOBEC-2*$^{\text{Cons}}$, *BCbox-1*$^{\text{Cons}}$ and *BCbox-2*$^{\text{Cons}}$. The presence of this combination suggests protection by identifying eleven sequences from Class 0 ($< 10,000$ virus copies/mL) and only referencing nine of the 49 sequences from Class 1 ($\geq 10,000$ virus copies/mL). Similarly, the second-best combination for VLIni defined by *APOBEC-2*$^{\text{Mut}}$, *BCbox-1*$^{\text{Cons}}$ and *BCbox-2*$^{\text{Mut}}$ achieved a p-value of 0.0415 and suggests risk since its presence has a high proportion of sequences of Class 1, only identifying two of the Class 0.

The only combination found as significant for VLHist was defined by *NLIS*$^{\text{Mut}}$, *BCbox-1*$^{\text{Cons}}$ and *BCbox-2*$^{\text{Mut}}$. This combination suggests risk since its presence identifies six sequences from Class 1 and only one from Class 0.

4.2 Discussion

Next are discussed the results found after applying the proposed methodology to the HIV-vif dataset. This data described genetic configurations and clinical statuses. The methodology makes a variable assessment considering coincidences among three machine learning algorithms (Apriori, MDR and C4.5). This is further complemented by an MLP assessment on the selected Apriori-rules, MDR-models and C4.5 tree-branches. The process started by defining the dataset encoding and analysing the variables. This allowed the definition of seventeen variables with four distinct outputs. After the analysis, the variable *APOBEC-1* was removed from the dataset since it had only one value (i.e. the variable is not informative). Then different Apriori-rules, C4.5 decision trees and MDR-models were generated by using the modified dataset. Next, a selection of the best results from each algorithm (rules, trees and models) was done. The selected rules, trees and models were then compared against a defined baseline MLP-classification. The baseline was calculated by applying the MLP training process on all the variables and per each output. For the comparisons the classification performance on using the variables present in each selected rule, tree or model as input along each output for the MLP training process was determined. Information from the results was combined using a previously described points system in order to determine the relevance per variable. The analysis of the suggested combinations from the algorithms allows determining an importance value for each variable. After calculating these values, the variables more related to each output considering the Class 1 ($<$500 CD4+ T cells/μL or $\geq$10,000 virus copies/mL for the CD4+ T or viral loads outputs respectively) were selected. The selected variables were then used in the generation of decision trees with ID3, from which were determined the hypotheses delivered as the final result of the methodology. Finally, these hypotheses were tested using Fisher's exact test. Seven of these were found associated with some of the outputs. The convenience of the use of our methodology is shown in the number of interesting hypotheses for assessing. Considering the removal of the *APOBEC-1* variable, a brute force approach on the dataset with sixteen variables would require testing 43,046,720 hypotheses per output, giving 172,186,880 hypotheses for the four outputs. On the other hand, the methodology allows the reduction to only 23 candidate hypotheses that reflect both the relevance from the variables and the interactions suggested as highly important.

Combining different approaches allows a better assessment of the importance of each variable in relation to each class. This is since each algorithm has a unique way of assessing variable relevance, and the coincidence among them adds support to the findings. On the other

hand, using the ID3 algorithm allows a reduction in the number of final specific combinations to explore by including only those suggested as more informative.

For example, by using the sets of three relevant variables identified for CD4Ini, VLIni and VLHist in ID3 allowed the identification of only seven, six and six combinations as relevant, respectively. On the contrary, exploring the total number of combinations would yield 52 tests per output considering Equation 1.1. A similar case was present for CD4Hist, where instead of testing ten combinations, only four were suggested as relevant.

Using Fisher's exact test with a significance level of $\alpha < 0.05$ allowed the assessment on the combinations. A total of seven different hypotheses were found as associated with the outputs. Of these combinations, four can be considered as protective while the remaining three can be regarded as risky. To the best of our knowledge, this is the first time that all these combinations are suggested.

4.2.1 Results considering the genetic variables

The best combination for CD4Ini includes the conservation of the *Cul5-3* segment of the vif sequence while having mutations on *APOBEC-4* and *BCbox-3*. These variables are described by the positions of the segments (160-169), (39 & 48) and (156-158), respectively. The combination suggested protection since it was not present in any of the sequences of Class 1 (<500 CD4+ T cells/μL). A more detailed analysis on the variables shows that the substitutions F39C and H48N were the most relevant mutations in the *APOBEC-4* variable, which were previously reported as important in the interaction with APOlipoprotein B messenger RNA Editing enzyme, Catalytic polypeptide-like (APOBEC3)H [12, 127, 121]. The principal mutation on *BCbox-3* was K158Q whose effects had not been identified, but it is considered relevant for cross-linking during oligomerisation[2] [33]. F39C, phenylalanine for cysteine at the vif-position 39, represents the change of non-polar amino acid for a polar one. H48N, histidine for asparagine, indicates the change of a positively charged chain for an uncharged one. This is the same case for the amino acid change K158Q, lysine for glutamine. These biochemically relevant changes suggest a declination in the virus function. The second-best hypothesis for CD4Ini also suggests protection by the presence of mutations on the variables *APOBEC-4*, *BCbox-3*, *Cul5-3*. The most important amino acid substitutions F39S, H48N for *APOBEC-4*, P156T and K158Q for *BCbox-3* and T167K/R for *Cul5-3*. Similarly to K158Q, the effects of the mutation P156T

[2]Oligomerisation describes the formation of a polymer molecule composed by a small number of monomers (i.e. smaller molecules).

has not been identified, but this is also part of a multimerisation domain[3] [180, 106]. T167 is important for formating the ligase complex ElonginB (EloB)-ElonginC (EloC)-Cullin5 (Cul5) E3 [181, 10]. F39S, phenylalanine for serine, and P156T, proline for threonine, involve the substitution of non-polar amino acid for a polar one. T167K/R, threonine for lysine or arginine, indicates the change of uncharged amino acid for one with a positively charged chain.

The two variables identified in the case of CD4Hist are related to the interaction vif-APOBEC3. The segments involved in these variables are (22-26) for *APOBEC-2* and (40-44) for *APOBEC-3*. The best hypothesis suggests risk by only being present in one sequence of Class 0 ($\geq$500 CD4+ T cells/μL) and 26 of Class 1. This imbalance is present when there are mutations in the segment for *APOBEC-2* while having *APOBEC-3* conserved. The relevant substitutions involved K22I/N, S23R and V25G. K22I, lysine for isoleucine, implies the change of a positively charged amino acid for one with a hydrophobic side chain, while K22N, lysine for asparagine, indicates the change for one uncharged amino acid. On the contrary, S23R, serine for arginine, indicates the change of uncharged amino acid for a positively charged one. Similarly, V25G, valine for glycine, involves the change of non-polar amino acid for a polar one. Mutations on K22 [24, 56] are relevant in the interaction with APOBEC3G while the $S^{23}LV^{25}$ motif is relevant in the degradation of APOBEC3F/G [161]. The presence of these mutations suggests an improvement in virus efficiency. On the contrary, the second hypothesis for CD4Hist suggests protection by identifying most of the sequences of Class 0. The absence of this combination, conservation on the *APOBEC-2* and *APOBEC-3* segments, is highly related with a better clinical status for the historical CD4+ T cells measurements.

The hypotheses related to the viral loads had the highest p-values, being in the range [0.0318-0.0415]. The best combination for VLIni was defined by conservation on the variables *APOBEC-2*, *BCbox-1* and *BCbox-2* corresponding to the vif segments (22-26), (144-149) and (151-154), respectively. This combination suggests protection since its presence is related to fewer cases from Class 1, $\geq 10,000$ virus copies/mL. On the other hand, its absence appears on most of the sequences having this class. On the contrary, the second-best combination suggests risk when the *APOBEC-2* and *BCbox-2* are mutated while *BCbox-1* is conserved. In this case, the presence of the combination is related with only two of the 26 sequences of Class 0, $<$10,000 virus copies/mL. The most common mutations for the *APOBEC-2* segment were the same than those for CD4Hist (K22I/N, S23R and V25G) combined with A151S/T and I154H/K/R/T for *BCbox-2*. A151 has been identified as relevant for proteolytic processing by viral protease [80] while mutations on I154 might affect the APOBEC3G degradation efficacy [14]. A151S/T,

[3]Multimerisation describes the formation of a protein molecule composed by two or more polypeptide chains.

alanine for serine or threonine, implies the change of hydrophobic amino acid for one with polar uncharged side chains, being the same case for I154T, isoleucine for threonine. Similarly, I154H/K/R, histidine for lysine or arginine, implies the change of hydrophobic amino acid for a positively charged one.

Finally, the only associated hypothesis for VLHist involved mutations on *APOBEC-2* and *NLIS*, while having *BCbox-1* conserved. This combination was found in six sequences from Class 1 and only in one from the other class, suggesting a greater risk. Again, the most relevant mutations for *APOBEC-2* were K22N and V25G, while in the *NLIS*-segment involved K91D/E and K92E. K91 and K92 are part of the motif RKKR, involved in ensuring the cytoplasmic localisation of vif since its inclusion in the cell's nucleus make its protein non-functional [45, 9]. The change of K for D/E, lysine for aspartic or glutamic acid, represents the replacement of a positively charged amino acid for another negatively charged.

4.2.2 Results considering the machine learning approaches

The use of our methodology has allowed the suggestion of specific combinations which can be used as hypotheses for statistical tests. This approach could allow a reduction in the workload for the experts while introducing novel interactions without using the impractical process of exploring so many combinations, which could render the results useless due to the necessary correction process.

Although the use of machine learning approaches such as Apriori and MDR might present difficulties given the number of variables, their use helps in complementing the variable assessment.

In the case of the Apriori algorithm, we found in our experimentations that it works well in a dataset without problems with less than 29 variables, even considering low values for the confidence parameter. On the other hand, the use of MDR in the exploration of models including six variables. However, exploring models with more variables is a complicated process that would represent a high computational load. The process with C4.5, on the other hand, is faster and less affected by the number of variables. However, the results suggest that assessing the variables is better when considering the three approaches compared. As mentioned before, the individual results from the algorithms do not give a clear indication on which are the more important variables. This is further improved by using Artificial Neural Networks (ANNs), allowing another point for comparison.

Other approaches have been used for assessing the variables on previous versions of the HIV-vif dataset. While some of the approaches had shown positive effects on the classification performance, it was similar to that from our proposed methodology when using as ANNs or Support Vector Machines (SVMs) (see Annexe C). Therefore, our proposed methodology covers both an adequate classification while allowing suggesting variable importance and specific combinations for hypothesis testing. The major difficulties faced on the experiments were related to the relatively small sample processes and the imbalance on the classes, mainly for the CD4+ T clinical status. On the other hand, the low classification performance on the outputs with measurements on the viral loads might be related to the need for more examples for learning.

Conclusions

In this work, we propose a new methodology for identifying important variable combinations. This methodology includes four machine learning approaches: association rules induction, Decision Tree Induction, data mining and Artificial Neural Network (ANN). Apriori, J48 for C4.5, and Multifactor Dimensionality Reduction (MDR) algorithms were used for variable assessment combined with Multi-Layer Perceptron (MLP). Finally, Iterative Dichotomiser 3 (ID3) helped in suggesting hypotheses for statistical testing. To the best of our knowledge, this is the first work associating complex combination of multidimensional variables representing viral infectivity factor (vif) gene segments and two clinical follow-up parameters. These measurements, Cluster of Differentiation 4 (CD4)+ T cells and viral load, are some of the most important indicators on the transition from the Human Immunodeficiency Virus (HIV) infection to Acquired Immune Deficiency Syndrome (AIDS).

HIV is still a worldwide problem and the search for strategies for combating the infection continue to be relevant. One of such strategies is the search for associations between genetic descriptors and some indicator of health status. This is a complex problem since the traditional approach for the exploration of these possible associations is through the suggestion of hypotheses by experts. The process is time-consuming and highly difficult since is based on a manual approach. On the other hand, the search for hypotheses is further complicated by the need of applying correction methods according to the number of tests applied. In this sense, making a high number of tests might end requiring a highly stringent p value. Considering these complications, in recent years there has been an increase in the application of machine learning methods for the search of associations. For those reasons, we proposed a new methodology for the search of associations. The convenience on the use of our machine learning methodology is that these methods can apply inductive learning on non-linear data without needing a model predefinition. The methods included in our experimentation can generate models mapping from the inputs (i.e. HIV-vif protein attributes) to the specific outputs (i.e. clinical endpoints) in the dataset.

The proposed methodology was tested on a new dataset generated from a previous work describing the HIV-vif segment from 68 Mexican-mestizo patients in San Luis Potosí, Mexico. The data contained 17 describing variables and four different clinical outputs, all of them with binary values (i.e. zero or one according to absence or presence, respectively). In this case for each output, there would be necessary to test 43,046,720 hypotheses considering only sixteen binary variables. When the four different clinical outputs are considered, the total number of hypothesis would amount to 172,186,880. On the contrary, the use of our methodology yielded only 23 hypotheses after the suggestion on the more relevant variables per output and interactions identification. Although our methodology does not eliminate the need for traditional statistical methods, it helps in identifying relevant factors and in suggesting important interactions among them. This, in turn, helps in guiding the identification of novel potential associations for testing.

The methodology helped in suggesting the more relevant variables for each output by considering selected rules, models and trees learned by the algorithms Apriori, MDR and C4.5. This was further enhanced by using MLPs for assessing classification improvement. The results from the algorithms (rules, trees and models) were used for the estimation of the variables influence for a better discrimination between classes. This assessment was based on determining a baseline classification performance considering the use of all the sixteen variables in the MLP training process. Then the variables included in each of the selected Apriori-rules, MDR-models and C4.5-tree-branches were tested as inputs on the MLP training process. This helped in determining which variables were being picked in the most discriminant combinations. This information was combined in a ranking method based on empirically determined weights for the results of each algorithm. The variable assessment was done according to their identification as relevant for each algorithm and their influence on the classification performance. The ranking allowed selecting the two most relevant variables for CD4Hist and three on the other outputs. The final step on the methodology involved generating specific hypotheses by inducting decision trees with ID3. Then 23 hypotheses were tested using Fisher's exact test. Seven of these hypotheses were found as associated with the outputs. Only the output VLHist had one, while each of the other outputs had two. The results from the Fisher's exact test were further supported by a logistic regression statistical test for independence on the seven combinations found as associated in all outputs but VLHist [3]. For this, the logistic regression test was performed using the non-stratified real value data from the outputs. The specific combinations tested corresponded to *BCbox-3*$^{\text{Mut}}$, *APOBEC-4*$^{\text{Mut}}$, *Cul5-3*$^{\text{Mut}}$ for CD4Ini, *APOBEC-2*$^{\text{Cons}}$, *APOBEC-3*$^{\text{Cons}}$ for CD4Hist and *APOBEC-2*$^{\text{Cons}}$, *BCbox-1*$^{\text{Cons}}$, *BCbox-2*$^{\text{Cons}}$ for VLIni.

The importance of the vif variables identified as more relevant for each clinical output is in agreement to previous findings related to the interaction with APOlipoprotein B messenger RNA Editing enzyme, Catalytic polypeptide-like (APOBEC3). The BCbox segments represented by the variables express functions relevant in the hijacking of the E3 ligase and its posterior attachment on the APOBEC3 proteins. The BCbox segments do this by interacting with ElonginC (EloC) and a zinc finger motif that connects with Cullin5 (Cul5). In this way, the vif protein can recognise APOBEC3 through Cul5 forming a complex with ElonginB (EloB) and EloC that leads to recruiting Core Binding Factor β (CBFb). The interaction between vif and EloC is mediated by the motif ^{144}SLQ(Y/F)LA^{149} identified by *BCbox-1* and that is part of the Elongin B/C box [181, 47]. Previous findings have shown the crucial role of the short side chain Ala^{149} in EloC binding. This domain is thought to play HIV-vif's most critical process in APOBEC3 suppression. Similarly, mutations on the APOBEC-2 region were associated with a higher risk for historic CD4+ T cells and historic viral loads. The results confirm that the variables suggested by our methodology can successfully serve as hypotheses in statistical tests. Nevertheless, these combinations require further research to truly assess their biological significance. This is due to the fact that finding a statistical interaction is not enough to guarantee their existence on a biological level.

The proposed methodology also allowed us to identify the most relevant variables in the dataset per clinical output. Combining distinct approaches allowed improving the results of using them separately. In the case of the Apriori algorithm, the main disadvantage is related to the limitation in the number of variables that can be analysed because of its high computational cost. A limitation of ID3 and C4.5 is that the selection in each step of the process is determined on the individual importance of the available variables, which ignores the direct influence of interactions and therefore reduce its utility in detecting epistasis [25]. They also suffer from the "selective superiority" which is having good performance in only some domains [37]. ID3 is also affected by a lower generalisation after training while being strongly biased to the cases with many outcomes. On the other hand, the use of C4.5 can result in highly complex trees which might complicate their interpretation. The MDR disadvantage is also related to the computational resources required for the analysis of more complex models (those with a higher number of variables, e.g. ten). As with the complex trees from ID3 and C4.5, the MDR models can also be of difficult interpretation. Additionally, MDR's search of models with a higher number of variables can cause a great number of empty cells if there are missing or singleton data [144]. The major disadvantage of using ANNs is the difficulty in extracting the most important factors that define the generated models. Furthermore, choosing the best parameters

and architecture for the training process is a difficult task. The methodology major limitation is the number of variables that can be used for the analysis due to the limitations of the Apriori algorithm and MDR. On the other hand, the approach allows identifying the most relevant variables or features while being capable of suggesting specific hypotheses.

The classification performance achieved using the different classification approaches might be considered as low. However, these results are related to the difficulties of learning from a small size dataset (only 77 sequences), which makes the learning process more difficult. On the other hand, the suitability of the methods can be seen in that their classification performance is similar to those methods in the state-of-the-art such as ANNs, Support Vector Machine (SVM) and Random Forest (RF). Although there was not a complete experimentation, initial trials using feature selection algorithms such as ReliefF [82], Synthetic Minority Over-sampling Technique (SMOTE) [120] and tensor theory [36] showed similar results as those from the use of our methodology in a previous version of the HIV-vif dataset when considering the CD4Ini output. Similarly, some initial trial using Principal Component Analysis (PCA) [131] showed no advantage from the suggested variables. Nevertheless, more research is needed for a better comparison between the methods and our methodology.

Future work

Future work can be divided considering the data and the changes to the methodology for identifying relevant variables. On the data side, an additional analysis could be made using different codifications for the original dataset. For example, the use of all the 192 amino acid positions as the input data. Other analysis could be made by using a different definition on the variables considering suspected relevant positions and mutations, for example, $K22H$. Additionally, other analysis could be made on a bigger dataset in the search of associations. This process could be applied on a more complex task such as searching interactions among the patients' genetic immune system data and clinical indicators on the HIV infection progression such as the presence of specific AIDS-related diseases. The host immune system information could be from genes such as Killer-cell Immunoglobulin-like Receptors (KIR) and Human Leucocyte Antigen (HLA). These additional analyses might be further supported by previous findings on these domains and further support the convenience of the use of our methodology.

On the other hand, modifications on the methodology can be explored looking for a better way of assessment on the variables. One way could be addressing the pertinence on using other approaches such as RF, Testors, Linear Discriminant Analysis and Principal Components

Analysis in the methodology. This might help in reducing the disadvantage of the number of variables that can be used or give more insight into the existing interactions.

Another possible change is related to the assessment of the weights applied to the algorithms results. In our methodology, the definition of said weights was made considering an empirical approach. This could be improved by using other Artificial Intelligence (AI) methods such as meta-heuristics. For example, an evolutionary approach could be used for identifying weight combinations that deliver a better estimation rank of the importance of the variables. One way of assessing these could be through the use of candidate combinations of weights values and additional values for testing the number of variables to include in the final rank. This process could be guided by the classification performance achieved with the top-ranked variables according to each solution. In this way, the evolved solutions might deliver a most appropriate set of weights for the problem, while helping in determine how many variables must be included in the rank.

The methodology could be further improved by including some strategy for dealing with the imbalance on the data. Different approaches could be used for this: undersampling the more represented class, oversampling the less represented class. Another approach could be generating artificial data with the use of techniques such as Synthetic Minority Oversampling Technique (SMOTE). Bagging could also be used for a better estimation of the models' performance with the possibility of more information from the variables influence. Finally, a study for determining the best measurement for the best classification assessment could be explored. This could involve the use of other metrics such as a weighted accuracy for addressing the data imbalance.

Finally, future work also includes the analysis and assessment of possible biological significance on the combinations suggested by our methodology.

Annexes

Annexe A

For the study of biological data, scientists mainly use statistical tests. Frequency comparisons between healthy controls and diseased patients use a test on contingency tables. In this way, it is possible to estimate how significant is the association between an input variable and one disease status measurement.

Fisher's exact test

Ronald Fisher and Joseph Irwin proposed almost simultaneously in the 1930s the Fisher's exact test, also called the Fisher-Irwin test [49, 48, 71]. They based the test on the hypergeometric distribution for determining if there are non-random associations between two categorical or nominal variables. "Exact" refers to the fact that the results come from probability theory instead of being approximated [143].

The hypergeometric formula considers X elements from a population of N trials with a characteristic A (i.e. success) and $N-X$ not having it (i.e. fails). Then from a sample of n chosen elements the probability of r of them having A is given by Equation A.1 [67]:

$$P(X=r) = \frac{\binom{X}{r}\binom{N-X}{n-r}}{\binom{N}{n}} = \frac{\left(\frac{X!}{r!(X-r)!}\right)\left(\frac{(N-X)!}{(n-r)!(N-X-n+r)!}\right)}{\frac{N!}{n!(N-n)!}} \tag{A.1}$$

The data is summarised into categories in a contingency table since each cell is called contingent. These cells contain the count for the category of one variable that also lies on a specific value of another variable. The basic calculation involves the frequency (i.e. the counts) on which the examples are in a specific category. The basic contingency table is of size 2×2, which includes counts on four categories. An example of this table is presented in Table A.1,

considering sampling a population of size N with c_1 objects with the characteristic A and c_2 with not having it (i.e. $not-A$). Then a sample of r_1 of these objects can be drawn and then determine how many of them have the characteristic A, identified as a. Similarly, b identifies the number of objects that are $not-A$ in r_1. Using Fisher's exact test on this table allows assessing the association between rows and columns based on the probability of having the specific values on each cell.

Table A.1. Example of a 2×2 contingency table considering a sample of N objects having a characteristic A or not (i.e. $not-A$). Then it can be calculated how many objects of a sample r_1 have or not the A characteristic. Using Fisher's exact test on this table allows assessing the association between rows and columns.

	A	Not-A	Total
In sample	a	b	r_1
Not in sample	c	d	r_2
	c_1	c_2	N

The probability of the cells presenting specific values is calculating considering the following facts as explained by Hoffman [67]. There are $\binom{N}{r_1}$ possible samples. Of these, $\binom{c_1}{a}$ is the number of ways of choosing A in a sample of size c_1, while $\binom{c_2}{b}$ is the number of ways of choosing $not-A$ in a sample size $N-c_1=c_2$. Since these are independent, there are $\binom{c_1}{a}\binom{c_2}{b}$ ways of choosing a As and b $not-A$s. Therefore, the probability of choosing a As $= \frac{\binom{c_1}{a}\binom{c_2}{b}}{\binom{N}{r_1}} = \frac{\frac{c!}{a!c!}\times\frac{c_2!}{b!d!}}{\frac{N!}{r_1!r_2!}}$, which can be expressed using factorials as:

$$\frac{c_1!c_2!r_1!r_2!}{N!a!b!c!d!} \tag{A.2}$$

The calculation on the probability is based on the assumptions that the individual observations are independent. A second assumption is that the marginal sums are fixed or "conditioned" [104]. These sums are represented by the values for r_1, r_2 the rows and c_1 and c_2 for the columns.

The null hypothesis in this test is that the relative proportions of one variable are independent of those from the second variable. This signifies that nothing is changing with the behaviour of the variables or, in other words, that the proportions can be considered the same even with different values on the variables.

The probability of the difference between the variables' proportions is then calculated considering all the combinations that have lower probabilities than the observed data. In the case of the usual two-tailed test, this is done by considering those probabilities from the

observed data and those more extreme cases. The summation involves adding the probabilities from using all possible 2×2 tables that could have been observed, considering the rows and columns values fixed.

If the null hypothesis holds, then it implies that the variable in the rows was not found related to the variable in the columns. That means that one variable outcome gives no information for the outcome from the other variable. If the null hypothesis is rejected, then the test signals the presence of an association between the variables without indicating its strength.

The next example was taken from the Freeman & Campell tutorial on the analysis of categorical data [51].

Table A.2. Table based on the example from Freeman & Campbell on the analysis of categorical data. The example tries to assess the association between improvement of some symptoms related to the intake of magnesium or a placebo.

	Magnesium	Placebo	Total
Improve	12	3	15
Not improve	3	4	17
	15	17	32

Substituting in Equation A.2 with the values a=12, b=3, c=3, d=14, r_1= 15, r_2=17 and c_1=15, c_2=17, the probability for the table corresponds to:

$p - value = \frac{15!17!15!17!}{32!12!3!3!14!} = 0.0005469$

For the calculation of the two-tail probability, the next step is to accumulate those probabilities being lower than the probability of a particular case. Table A.3 shows in bold the probabilities that need to be included in the summation for the particular case represented by the subtable *xiii*. On the other hand, the subtables *i*, *ii*, *iii*, *xiv*, *xv* and *xvi* show the cases with a lower probability that the particular case. Therefore, the two-tail probability after summing all these cases is p=0.001033. If the α value for significance were 0.05, then we would reject the null hypothesis and accept the alternative hypothesis. This would indicate that there exists an association between the rows and columns.

Table A.3. Sixteen different rearranging ways for Table A.2 while keeping the marginal sums fixed. The numbers in bold correspond to the probabilities needed for calculating the two-tail probability. These are lower than the probability from the case of interest (subtable xiii).

	i		ii		iii		iv	
	0	15	1	14	2	13	3	12
	15	2	14	3	13	4	12	5
p-values	**0.0000002**		**0.0000180**		**0.0004417**		0.0049769	

	v		vi		vii		viii	
	4	11	5	10	6	9	7	8
	11	6	10	7	9	8	8	9
p-values	0.0298613		0.1032349		0.2150728		0.2765221	

	ix		x		xi		xii	
	8	7	9	6	10	5	11	4
	7	10	6	11	5	12	4	13
p-values	0.2212177		0.1094916		0.0328475		0.0057426	

	xiii		xiv		xv		xvi	
	12	3	13	2	14	1	15	0
	3	14	2	15	1	16	0	17
p-values	***0.0005469***		**0.0000252**		**0.0000005**		**0.0000000**	

Annexe B

This annexe explains the process for determining the type of data encoding and number of features for the HIV-vif dataset.

Process for the HIV-vif Dataset definition

The original dataset generated by Guerra-Palomares et al. contained 192 variables, according to the HXB2 consensus sequence reference (GenBank: K03455.1). Each of these corresponds to the sites coding for the Human Immunodeficiency Virus (HIV)'s viral infectivity factor (vif) protein [56]. After a posterior definition of specific segments and sites by the *Laboratorio de Genómica Viral y Humana*, new variables were determined. These segments were combined with the information of four different outputs, as defined by Cluster of Differentiation 4 (CD4)+ T cells and viral counts. These outputs are described in Section 1.4.

A different number of variables were tested along with two data types, "binary" and "numeric". The former indicated if one or more mutations were presented in a specific interest region. The latter indicated the summed number of these mutations. For example, segment ^{22}KSLVK^{26} indicates the amino acids that are normally present according to the HXB2 reference sequence. Therefore, if a sample has the values ^{22}N–G-^{26}, then it indicates two relevant mutations. If the data type for the variables is "numeric", then the value for the variable is set as "2". On the other hand, if the data type is "binary", then the variable is set to "1", indicating that there exists at least one mutation on the sequence.

Three different schemes were considered concerning the number of variables: "all-vars", "no-totals" and "only-totals". The scheme "no-totals" was integrated by seventeen variables:

- Eight APOlipoprotein B messenger RNA Editing enzyme, Catalytic polypeptide-like (APOBEC3) variables represent segments related with the interaction of vif with the proteins APOBEC3C/D/F/G/H: APOBEC-1 (segment 14-17), APOBEC-2 (segment 22-26), APOBEC-3 (segment 40-44), APOBEC-4 (sites 39 and 48), APOBEC-5 (segment

69-72), APOBEC-6 (segment 74-78), APOBEC-7 (segment 81-88), APOBEC-8 (segment 171-175).

- Two variables pertain to vif segments that allow bounding with Core Binding Factor β (CBFb) to maintain stable vif expression levels and virus infectivity [9]: CBFb-1 (segment 84-89), CBFb-2 (segment 102-107).
- One variable reflects the changes in the segment corresponding to the Nuclear Localisation Inhibitory Signal (NLIS) (segment 90-93) important for cytoplasmic localisation in the target cell of the vif protein by interfering with the nuclear import pathway.
- Three variables relate with elongin B/C-box binding motif Bcbox-1 (segment 144-149), Bcbox-2 (segment 151-154), and Bcbox-3 (segment 156-158).
- The sites relevant for the Cullin binding motif and box: Cul5-1 (segment 120-121), Cul5-2 (site 124), Cul5-3 (segment 160-169).

The "all-vars" scheme considered the same variables as the "no-totals" plus four more. These represented the accumulated values from specific mutations as listed below:

- APOBEC-total represented the accumulation from the variables APOBEC1 to APOBEC8.
- CBFb-total represented the accumulation from the variables CBFb1 and CBFb2.
- Bcbox-total represented the accumulation from the variables Bcbox-1 to Bcbox-3.
- Cul5-total represented the accumulation from the variables Cul5-1 to Cul5-3.

The "only-totals" scheme considered only the totals, those defined in the scheme "all-vars", including the values for the variable NLIS-total.

Two different approaches were applied for evaluating the best data encoding and the groups of variables. The first was by considering the accuracy values determined from the algorithms C4.5 [139] and Multifactor Dimensionality Reduction (MDR) [144], in both cases considering a 10-Cross-Validation (CV). The second approach considered evaluating the accuracy obtained on specific combinations from the C4.5 decision trees and MDR-models by using the Multi-Layer Perceptron (MLP) method, a type of Artificial Neural Network (ANN). The implementations by Hall et al. in Weka were used for the algorithms C4.5 (named J48) and MLP. Similarly, the MDR implementation by Hanh et al. was used for the model generation [60].

Table B.1 shows the average accuracy on each output on each data type. This comparison used the classification performance from the decision trees generated by C4.5 with the Confidence Level (CF) values in the range [0.01-0.51]. In the same way, the classification performance from the MDR models up to six variables was considered. Since the "only-totals" scheme only had five variables, only five MDR-models generated with this scheme. These results, shown as *Only C4.5/MDR* on the table, exhibit a similar or better performance using "binary" data type with C4.5. With the MDR-models, all the results were better using the "binary" data type. The tests by combining the specific variables in the decision trees branches and those indicated by the MDR-models (shown as *Algorithms + MLP*) exhibit a similar trend. Each combination was tested as input on the MLP algorithm. The average values from these test show that the "binary" data type had similar or better performance than using "numeric" encoding. Only in the case of CD4Hist, the performance using the data type "numeric" was much better.

Table B.1. Average classification accuracy per output on two data types: "binary" and "numeric". The results are from two different approaches: i) the use of C4.5 or MDR and, ii) the use of the decision trees' branches or MDR-models as inputs for MLP training.

Approach	Output	Data type	C4.5	MDR	Output	Data type	C4.5	MDR
Only C4.5/MDR	CD4Hist	Binary	75.71	65.52	CVHist	Binary	56.0	46.8
		Numeric	75.52	55.69		Numeric	60.87	44.46
	CD4Ini	Binary	78.18	65.52	CVIni	Binary	59.14	58.21
		Numeric	76.88	55.69		Numeric	60.11	55.6
Algorithms + MLP	CD4Hist	Binary	52.88	77.06	CVHist	Binary	62.06	61.9
		Numeric	77.69	74.89		Numeric	61.56	61.9
	CD4Ini	Binary	78.4	76.19	CVIni	Binary	66.13	61.04
		Numeric	78.94	78.79		Numeric	63.74	64.07

The comparison among the variables schemes is shown in Table B.2. The results show that using the group "no-totals" had similar performance when considering the results of the algorithms C4.5 and MDR. This was also the case when the combination of the trees and models were used as input for the MLP algorithm. Only in the case of CD4Hist, the use of the group "only-totals" showed a very bad performance.

Considering these results, the dataset HIV-vif was created including the group "no-totals" and the binary codification. This is further supported by the fact that using a numeric codification

Table B.2. Average classification accuracy per output on three tested schemes: “all-vars”, “no-totals” and “only-totals”. The results are from two different approaches: i) the use of C4.5 or MDR and, ii) the use of the decision trees’ branches or MDR-models as inputs for MLP training.

Approach	Output	Variables	C4.5	MDR	Output	Variables	C4.5	MDR
Only C4.5/MDR	CD4Hist	All variables	75.16	59.56	CVHist	All variables	58.85	43.88
		No totals	75.63	62.03		No totals	58.57	46.91
		Only totals	77.27	60.15		Only totals	60.17	46.44
	CD4Ini	All variables	76.69	59.56	CVIni	All variables	60.13	60.63
		No totals	77.92	62.03		No totals	60.59	60.29
		Only totals	78.35	60.15		Only totals	57.79	48.37
Algorithms + MLP	CD4Hist	All-vars	78.44	78.57	CVHist	All-vars	61.21	62.34
		No-totals	78.44	75.32		No-totals	61.96	63.64
		Only-totals	38.96	74.03		Only-totals	62.26	59.74
	CD4Ini	All-vars	78.65	77.27	CVIni	All-vars	66.61	65.58
		No-totals	78.23	75.97		No-totals	64.62	58.44
		Only-totals	79.13	79.22		Only-totals	63.58	63.64

produces decision trees and MDR-models that are more difficult for interpretation. This is because using variables with a higher number of values produces trees with more branches. The same happens with the MDR-models, where there are generated more cells. On the other hand, using the scheme “all-vars” produces a redundancy in the data. On the contrary, using the scheme “only-totals” produces decision trees and MDR-models without much detail, making more difficult the identification of the relevant segments.

Annexe C

This annexe presents a comparison considering one of the algorithms involved in the methodology (C4.5 [139]) and other methods, including some considered as state-of-the-art. This comparison was made on a more complex version of the Human Immunodeficiency Virus (HIV)-viral infectivity factor (vif) dataset, containing 70 variables and 75 sequences. Different statistical non-parametric tests were applied for the comparison, including the Friedman test and the Quade Test. The results show that there is not a statistically significant difference in performance between C4.5 and the algorithm with the best on this dataset.

Classification comparison

Since the classification performance achieved with the methodology could be considered as low, a comparison against some state-of-the-art classifiers was made. The comparison considers a more difficult problem using more variables generated from the original HIV's vif-data (see chapter 1). Nevertheless, the variables represent genetic segments of the vif gene and two of the clinical statuses, CD4Ini and CD4Hist. An example of the dataset is shown in Table C.1.

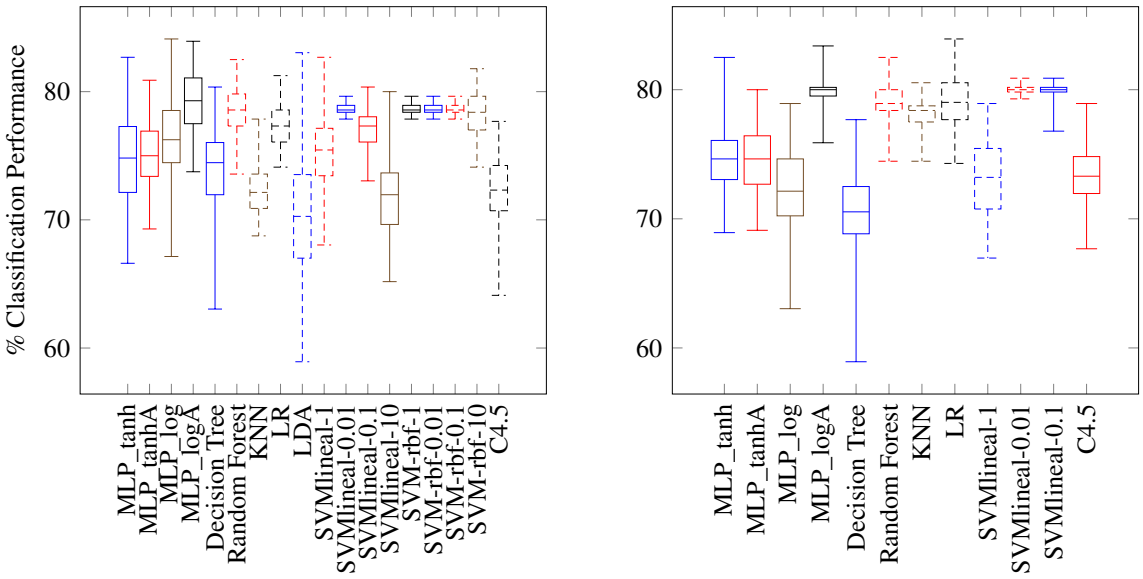
In this case, only the algorithm C4.5 [139] implemented in Weka [61] was compared against other algorithms, some of which considered as state-of-the-art. The selected algorithms were: Classification and Regression Trees (CART) [18], *K* Nearest Neighbours (KNN) [31], Linear Discriminant Analysis (LDA) [64] §4.3, Logistic Regression (LR) [64] §4.4, Multi-Layer Perceptron (MLP) [152], Random Forest [17], and Support Vector Machine (SVM) [30]. All the algorithms were used with their implementation by Pedregosa et al. [132]. The default configuration was used, except in the MLP and SVM classifiers. In the case of MLP the parameter training iterations was set at 10,000, and only one hidden layer with 138 neurons was used. Two different activation functions were employed: hyperbolic tangent (MLP_tanh) and logistic sigmoid (MLP_log). Additionally, two different trained methods were used: Adam [81], a modification of the stochastic gradient descend, and Limited-memory Broy-

Table C.1. Example of the variables and outputs in the dataset.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	…	69	70		
Sequence	**W**	**DRMR**	**K22**	**K22N**	**K22H**	**V25G**	**F39**	**NP39**	**F39S**	**F39C**	**H48**	**H48N**	**R41**	**R41K**	**H43P**	**L/V72**	…	**T170**	**EDRWN**	**CD4Ini**	**CD4Hist**
s_1	1	1	0	1	0	1	1	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_2	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	0	1
s_3	1	1	0	1	0	0	0	1	0	0	0	1	1	0	0	1	…	1	1	0	1
s_4	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	1	1
s_5	0	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	1	1
s_6	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	1	1
s_7	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	1	1
s_8	1	1	1	0	0	0	0	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_9	1	1	1	0	0	0	0	0	1	0	1	0	1	0	0	1	…	1	1	1	1
s_{10}	1	0	0	1	0	0	0	1	0	0	0	1	1	0	0	1	…	1	1	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
s_{66}	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	1	…	1	1	1	1
s_{67}	1	1	1	0	0	0	0	0	0	1	0	1	1	0	0	1	…	1	1	1	1
s_{68}	1	1	0	1	0	0	0	1	0	0	1	0	0	1	0	1	…	1	1	1	1
s_{69}	1	1	1	0	0	0	1	1	0	0	1	0	1	0	0	1	…	1	1	0	0
s_{70}	1	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1	…	1	1	0	0
s_{71}	1	1	0	1	0	1	0	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_{72}	1	1	0	1	0	0	0	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_{73}	1	1	0	1	0	0	1	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_{74}	1	1	0	1	0	0	1	1	0	0	1	0	1	0	0	1	…	1	1	1	1
s_{75}	1	1	0	1	0	0	1	1	0	0	1	0	1	0	0	1	…	1	1	1	1

den–Fletcher–Goldfarb–Shanno (L-BFGS) [5]. Therefore, the MLP classifiers trained with the Adam method are identified as MLP_tanhA or MLP_logA, otherwise, the L-BFGS was used. In the case of SVMs, two different kernels were used: linear and Radial Basis Function (RBF). Additionally, the parameter C was set as 0.01, 0.1, 1, and 10. The medians of the classification performance (accuracy) of 100 training process are shown in Table C.2. From these results, it is visible that most of the algorithms had a similar classification, being MLP_logA the best and SVMlineal-10, LDA, and KNN the ones with the lowest values (see Figure 4.9).

A statistical analysis was carried on to evaluate the null hypothesis which is: there is no difference in performance among the algorithms (H_0). The alternative hypothesis is that at least one algorithm has a significant difference in performance (H_1). For all the tests, values with a p-value smaller than 0.05 were considered as significant. The non-parametric tests Friedman [27] and Quade [136] were used to evaluate the hypothesis. All the test were made using the R package *Pairwise Multiple Comparison of Mean Ranks Package* [134]. The results



(a) *CD4Ini* (b) *CD4Hist*

Figure 4.9. Classification performance comparison among algorithms Decision Tree, KNN, C4.5, LDA, LR, MLP, Random Forest, SVM. (a) Classification performance comparison with the output *CD4Ini*. Only the performances of 18 of the 19 algorithms are shown. (b) Classification performance comparison with the output *CD4Hist*. Only the performances of twelve of the 19 algorithms are shown.

from the omnibus test are shown in Table C.3 and indicate that there exists evidence by both tests to reject the null hypothesis. The p-values were 0.0135 for the Friedman test and 0.0003 for the Quade test.

Since there was evidence that at least one of the algorithms had a significant difference in performance, a posterior pairwise comparison was made. For this, the algorithm having the best performance was used as control (MPL_logA) and was compared with the rest of the algorithms. Four different statistical tests were applied:

- Demsar's many-to-one test [34],
- Eisinga-Heskes-Pelzer and Grotenhuis many-to-one test [41],
- Nemenyi-Wilcoxon-Wilcox-Miller many-to-one test [68],
- Quade NxN test with TDist approximation [136]

Table C.2. Comparison of classification accuracy among classifiers sorted by their performance on the CD4Ini output. The results show that the best algorithm was MLP_logA, while the worst was Naive-Bayes.

Algorithm	CD4Ini	CD4Hist
MLP_logA	79.29	80
SVMlineal-0.01	78.57	80
SVM-rbf-1	78.57	80
SVM-rbf-0.01	78.57	80
SVM-rbf-0.1	78.57	80
SVM-rbf-10	78.40	79.38
Random Forest	78.57	78.93
SVMlineal-0.1	77.32	80
LR	77.32	79.02
KNN	72.14	78.39
MLP_tanhA	75	74.64
MLP_tanh	74.82	74.64
SVMlineal-1	75.45	73.21
MLP_log	76.25	72.14
C4.5	72.32	73.30
Decision Tree	74.46	70.54
SVMlineal-10	71.96	63.93
LDA	70.27	55.36
Naive-Bayes	55.80	40.89

Table C.3. Results of applying the test omnibus with the Friedman and the Quade tests, considering the classification performance on the outputs CD4Ini and CD4Hist.

	Friedman Test	Quade Test
p-value	0.0135	0.0003

As in the previous tests, the null hypothesis (H_0) is that there is not a difference in performance between the best classifier and the other algorithms, considering pairwise comparisons. The alternative hypothesis is that there exists a difference in performance between the best classifier and the rest of the algorithms, considering pairwise comparisons. The results from these tests are shown in Table C.4.

The results of the posthoc analysis suggest that MPL_logA had significant differences when compared with algorithms with the lowest performance. These were Naive-Bayes, LDA, and

Table C.4. Post-hoc analysis using MPL_logA as control and applying four different statistical tests. The results show that there is no evidence to reject the null hypothesis that MLP_logA and C4.5 had a similar performance.

	Post-hoc analysis using MPL_logA as control			
Algorithm	Demsar's many-to-one	Eisinga-Heskes-Pelzer and Grotenhuis many-to-one	Nemenyi-Wilcoxon-Wilcox-Miller many-to-one	Quade NxN test with TDist approximation
SVMlineal-0.01	1	1	1	1
SVM-rbf-1	1	1	1	1
SVM-rbf-0.01	1	1	1	1
SVM-rbf-0.1	1	1	1	1
SVM-rbf-10	1	1	0.998	1
Random Forest	1	1	0.999	1
SVMlineal-0.1	1	1	1	1
LR	1	1	0.974	1
KNN	1	0.9418	0.432	1
MLP_tanhA	1	1	0.602	1
MLP_tanh	1	1	0.531	1
SVMlineal-1	1	1	0.497	0.766
MLP_log	1	1	0.497	0.608
C4.5	0.662	0.5602	0.316	0.482
Decision Tree	0.422	0.3081	0.223	0.15
SVMlineal-10	0.158	0.0646	0.099	0.047
LDA	0.092	0.0215	0.062	0.024
Naive-Bayes	0.052	0.0046	0.038	0.012

for one of the tests, SVMlineal-10. However, the comparison with C4.5 indicates that there is not enough evidence to determine a significant difference in performance with MPL_logA. This was the case for the results from the four different tests. Therefore, in this case, there is not enough evidence to reject the null hypothesis that both algorithms, MLP_logA and C4.5, had a similar performance on this specific dataset. Although these results might lack the statistical power for a definitive conclusion, they suggest that the performance between these algorithms is similar.

References

[1] Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93*. ACM Press.

[2] Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, page 487–499, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[3] Altamirano-Flores, J. S., Guerra-Palomares, S. E., Hernandez-Sanchez, P. G., Ramirez-Garcialuna, J. L., Arguello-Astorga, J. R., Noyola, D. E., Cuevas-Tello, J. C., and Garcia-Sepulveda, C. A. (2020). Identification of HIV-1 vif protein attributes associated with CD4 t cell numbers and viral loads using artificial intelligence algorithms. *IEEE Access*, 8:87214–87227.

[4] Auffray, C. and Hood, L. (2012). Editorial: Systems biology and personalized medicine - the future is now. *Biotechnology Journal*, 7(8):938–939.

[5] Battiti, R. (1992). First- and second-order methods for learning: Between steepest descent and newton's method. *Neural Computation*, 4(2):141–166.

[6] Beam, A. L., Motsinger-Reif, A. A., and Doyle, J. (2014). Bayesian neural networks for detecting epistasis in genetic association studies. *BMC Bioinformatics*, 15(1):368.

[7] Becker, A. (2019). Artificial intelligence in medicine: What is it doing for us today? *Health Policy and Technology*.

[8] Belatreche, A., Maguire, L. P., Mcginnity, M., and Wu, Q. X. (2003). An Evolutionary Strategy for Supervised Training of Biologically Plausible Neural Networks. In *Proc. Sixth Int. Conf. Comput. Intell. Nat. Comput.*, pages 1524–1527.

[9] Bell, C. M., Connell, B. J., Capovilla, A., Venter, W. D., Stevens, W. S., and Papathanasopoulos, M. A. (2007). Molecular characterization of the HIV type 1 subtype c accessory genes vif, vpr, and vpu. *AIDS Research and Human Retroviruses*, 23(2):322–330.

[10] Bergeron, J. R. C., Huthoff, H., Veselkov, D. A., Beavil, R. L., Simpson, P. J., Matthews, S. J., Malim, M. H., and Sanderson, M. R. (2010). The SOCS-box of HIV-1 vif interacts with ElonginBC by induced-folding to recruit its cul5-containing ubiquitin ligase complex. *PLoS Pathogens*, 6(6):e1000925.

[11] Beyerer, J., Richter, M., and Nagel, M. (2017). *Pattern Recognition*. Walter de Gruyter GmbH, Berlin/Boston.

[12] Binka, M., Ooms, M., Steward, M., and Simon, V. (2011). The activity spectrum of vif from multiple HIV-1 subtypes against APOBEC3g, APOBEC3f, and APOBEC3h. *Journal of Virology*, 86(1):49–59.

[13] Bisaso, K. R., Anguzu, G. T., Karungi, S. A., Kiragga, A., and Castelnuovo, B. (2017). A survey of machine learning applications in HIV clinical research and care. *Computers in Biology and Medicine*, 91:366–371.

[14] Bizinoto, M. C., Yabe, S., Leal, É., Kishino, H., de Oliveira Martins, L., de Lima, M. L., Morais, E. R., Diaz, R. S., and Janini, L. M. (2013). Codon pairs of the HIV-1 vif gene correlate with CD4+ t cell count. *BMC Infectious Diseases*, 13(1).

[15] Boaz, M. J., Waters, A., Murad, S., Easterbrook, P. J., and Vyakarnam, A. (2002). Presence of HIV-1 Gag-Specific IFN- +IL-2+ and CD28+IL-2+ CD4 T Cell Responses Is Associated with Nonprogression in HIV-1 Infection. *Journal of Immunology*, 169(11):6376–6385.

[16] Boutorh, A. and Guessoum, A. (2016). Complex diseases SNP selection and classification by hybrid Association Rule Mining and Artificial Neural Network—based Evolutionary Algorithms. *Engineering Applications of Artificial Intelligence*, 51:58–70.

[17] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

[18] Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. CRC Press, New York.

[19] Bruce G. Buchanan, E. H. S. (1984). *Rule Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project (The Addison-Wesley series in artificial intelligence)*. The Addison-Wesley series in artificial intelligence. Addison-Wesley.

[20] Buchanan, B. G. and Feigenbaum, E. A. (1978). Dendral and meta-dendral: Their applications dimension. *Artificial Intelligence*, 11(1-2):5–24.

[21] Burton, P. R., Tobin, M. D., and Hopper, J. L. (2005). Key concepts in genetic epidemiology. *Lancet*, 366(9489):941–951.

[22] Centers for Disease Control and Prevention (2020). Diagnoses of HIV infection in the United States and dependent areas, 2018 (updated). HIV Surveillance Report; 31.

[23] Centro Nacional para la Prevención y el Control del VIH y el sida (2018). *Vigilancia Epidemiológica de casos de VIH/SIDA en México Registro Nacional de Casos de SIDA (Actualización al 09 de Noviembre de 2018)*.

[24] Chen, G., He, Z., Wang, T., Xu, R., and Yu, X.-F. (2009). A patch of positively charged amino acids surrounding the human immunodeficiency virus type 1 vif SLVx4yx9y motif influences its interaction with APOBEC3g. *Journal of Virology*, 83(17):8674–8682.

[25] Christmas, J., Keedwell, E., Frayling, T. M., and Perry, J. R. (2011). Ant colony optimisation to identify genetic variant association with type 2 diabetes. *Information Sciences*, 181(9):1609–1622.

[26] Coloccini, R. S., Dilernia, D., Ghiglione, Y., Turk, G., Laufer, N., Rubio, A., Socías, M. E., Figueroa, M. I., Sued, O., Cahn, P., Salomón, H., Mangano, A., and Pando, M. Á. (2014). Host genetic factors associated with symptomatic primary HIV infection and disease progression among argentinean seroconverters. *PLoS ONE*, 9(11):e113146.

[27] Conover, W. J. and Iman, R. L. (1981). Rank transformations as a bridge between parametric and nonparametric statistics. *The American Statistician*, 35(3):124–129.

[28] Cordell, H. J. (2009). Detecting gene–gene interactions that underlie human diseases. *Nature Reviews Genetics*, 10(6):392–404.

[29] Cordell, H. J. and Clayton, D. G. (2005). Genetic association studies. *Lancet*, 366(9491):1121–1131.

[30] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

[31] Cover, T. M. and Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

[32] Cuevas-Tello, J. C., Hernández-Ramírez, D., and García-Sepúlveda, C. A. (2013). Support vector machine algorithms in the search of KIR gene associations with disease. *Computers in Biology and Medicine*, 43(12):2053–2062.

[33] da Costa, K. S., Leal, E., dos Santos, A. M., e Lima, A. H. L., Alves, C. N., and Lameira, J. (2014). Structural analysis of viral infectivity factor of HIV type 1 and its interaction with a3g, EloC and EloB. *PLoS ONE*, 9(2):e89116.

[34] Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

[35] Dinov, I. D. (2018). *Data Science and Predictive Analytics*. Springer International Publishing.

[36] Dmitriev, A., Zhuravlev, Y., and Krendeliev, F. (1966). About mathematical principles and phenomena classification. *Diskretni Analiz*, (7):3–15.

[37] Domingos, P. (1996). Unifying instance-based and rule-based induction. *Machine Learning*, 24(2):141–168.

[38] Douek, D. C., Brenchley, J. M., Betts, M. R., Ambrozak, D. R., Hill, B. J., Okamoto, Y., Casazza, J. P., Kuruppu, J., Kunstman, K., Wolinsky, S., Grossman, Z., Dybul, M., Oxenius, A., Price, D. A., Connors, M., and Koup, R. A. (2002). HIV preferentially infects HIV-specific CD4+ T cells. *Nature*, 417(6884):95–98.

[39] Dunham, M. H. (2003). *Data mining introductory and advanced topics*. Prentice Hall/Pearson Education, Upper Saddle River, N.J.

[40] Eholié, S. P., Badje, A., Kouame, G. M., N'takpe, J.-B., Moh, R., Danel, C., and Anglaret, X. (2016). Antiretroviral treatment regardless of CD4 count: the universal answer to a contextual question. *AIDS Research and Therapy*, 13(1).

[41] Eisinga, R., Heskes, T., Pelzer, B., and Grotenhuis, M. T. (2017). Exact p-values for pairwise comparison of friedman rank sums, with application to comparing classifiers. *BMC Bioinformatics*, 18(1).

[42] Espinal, A., Carpio, M., Ornelas, M., Puga, H., Melin, P., and Sotelo-Figueroa, M. (2014). Comparing Metaheuristic Algorithms on the Training Process of Spiking Neural Networks. In Castillo, O., Melin, P., Pedrycz, W., and Kacprzyk, J., editors, *Recent Adv. Hybrid Approaches Des. Intell. Syst.*, volume 547 of *Studies in Computational Intelligence*, pages 391–403. Springer International Publishing.

[43] Fang, Y. H. and Chiu, Y. F. (2012). SVM-based generalized multifactor dimensionality reduction approaches for detecting gene-gene interactions in family studies. *Genetic Epidemiology*, 36(2):88–98.

[44] Farahani, M., Novitsky, V., Wang, R., Bussmann, H., Moyo, S., Musonda, R. M., Moeti, T., Makhema, J. M., Essex, M., and Marlink, R. (2016). Prognostic value of HIV-1 RNA on CD4 trajectories and disease progression among antiretroviral-naive HIV-infected adults in botswana: A joint modeling analysis. *AIDS Research and Human Retroviruses*, 32(6):573–578.

[45] Farrow, M. A., Somasundaran, M., Zhang, C., Gabuzda, D., Sullivan, J. L., and Greenough, T. C. (2005). Nuclear localization of HIV type 1 vif isolated from a long-term asymptomatic individual and potential role in virus attenuation. *AIDS Research and Human Retroviruses*, 21(6):565–574.

[46] Fauw, J. D., Ledsam, J. R., Romera-Paredes, B., Nikolov, S., Tomasev, N., Blackwell, S., Askham, H., Glorot, X., O'Donoghue, B., Visentin, D., van den Driessche, G., Lakshminarayanan, B., Meyer, C., Mackinder, F., Bouton, S., Ayoub, K., Chopra, R., King, D., Karthikesalingam, A., Hughes, C. O., Raine, R., Hughes, J., Sim, D. A., Egan, C., Tufail, A., Montgomery, H., Hassabis, D., Rees, G., Back, T., Khaw, P. T., Suleyman, M., Cornebise, J., Keane, P. A., and Ronneberger, O. (2018). Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature Medicine*, 24(9):1342–1350.

[47] Feng, Y., Baig, T. T., Love, R. P., and Chelico, L. (2014). Suppression of APOBEC3-mediated restriction of HIV-1 by vif. *Frontiers in Microbiology*, 5.

[48] Fisher, R. A. (1935a). *The Design of Experiments*. Oliver and Boyd, Edinburg.

[49] Fisher, R. A. (1935b). The logic of inductive inference. *Journal of the Royal Statistical Society*, 98(1):39.

[50] Fraser, C., Lythgoe, K., Leventhal, G. E., Shirreff, G., Hollingsworth, T. D., Alizon, S., and Bonhoeffer, S. (2014). Virulence and pathogenesis of HIV-1 infection: An evolutionary perspective. *Science*, 343(6177):1243727–1243727.

[51] Freeman, J. and Campbell, M. (2007). The analysis of categorical data: Fisher's exact test. *Scope*, 16:11–12.

[52] Gao, C., Sun, H., Wang, T., Tang, M., Bohnen, N. I., Müller, M. L. T. M., Herman, T., Giladi, N., Kalinin, A., Spino, C., Dauer, W., Hausdorff, J. M., and Dinov, I. D. (2018). Model-based and model-free machine learning techniques for diagnostic prediction and classification of clinical outcomes in parkinson's disease. *Scientific Reports*, 8(1).

[53] Goila-Gaur, R. and Strebel, K. (2008). HIV-1 vif, APOBEC, and intrinsic immunity. *Retrovirology*, 5(1):51.

[54] Gray, K. M., Valverde, E. E., Tang, T., e Alam Siddiqi, A., and Hall, H. I. (2015). Diagnoses and prevalence of HIV Infection Among Hispanics or Latinos — United States, 2008–2013. *MMWR. Morbidity and Mortality Weekly Report*, 64(39):1097–1103.

[55] Guerra Palomares, S. E. (2016). *Caracterización molecular de la proteína Vif del HIV-1 en México*. PhD thesis, Facultad de Medicina, Universidad Autónoma de San Luis Potosí.

[56] Guerra-Palomares, S. E., Hernandez-Sanchez, P. G., Esparza-Pérez, M. A., Arguello, J. R., Noyola, D. E., and García-Sepúlveda, C. A. (2015). Molecular Characterization of Mexican HIV-1 Vif Sequences. *AIDS Research and Human Retroviruses*, 31(00):aid.2015.0290.

[57] Guerrero, S., Libre, C., Batisse, J., Mercenne, G., Richer, D., Laumond, G., Decoville, T., Moog, C., Marquet, R., and Paillart, J.-C. (2016). Translational regulation of APOBEC3g mRNA by vif requires its 5′UTR and contributes to restoring HIV-1 infectivity. *Scientific Reports*, 6(1).

[58] Günther, F., Wawro, N., and Bammann, K. (2009). Neural networks for modeling gene-gene interactions in association studies. *BMC Genetics*, 10(1):87.

[59] Hagan, M. T., Demuth, H. B., Beale, M. H., and De Jess, O. (2014). *Neural Network Design*. Brooks/Cole, USA, 2nd edition.

[60] Hahn, L. W., Ritchie, M. D., and Moore, J. H. (2003). Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions. *Bioinformatics*, 19(3):376–382.

[61] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

[62] Han, B., Chen, X. W., Talebizadeh, Z., and Xu, H. (2012). Genetic studies of complex human diseases: characterizing SNP-disease associations using Bayesian networks. *BMC Systems Biology*, 6 Suppl 3(Suppl 3):S14.

[63] Hardin, J., Waddell, M., Page, C. D., Zhan, F., Barlogie, B., Shaughnessy, J., and Crowley, J. J. (2004). Evaluation of Multiple Models to Distinguish Closely Related Forms of Disease Using DNA Microarray Data: an Application to Multiple Myeloma. *Statistical Applications in Genetics and Molecular Biology*, 3(1):1–21.

[64] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York, New York, NY.

[65] Haykin, S. S. (2009). *Neural networks and learning machines*. Prentice Hall/Pearson, New York, 3rd. edition.

[66] Hinton, G. E., Osindero, S., and Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–54.

[67] Hoffman, J. I. (2015). Hypergeometric distribution. In *Biostatistics for Medical and Biomedical Practitioners*, pages 179–182. Elsevier.

[68] Hollander, M., Wolfe, D. A., and Chicken, E. (2014). *Nonparametric Statistical Methods. 3rd ed.* Wiley, New York.

[69] Huang, C., Mezencev, R., McDonald, J. F., and Vannberg, F. (2017). Open source machine-learning algorithms for the prediction of optimal cancer drug therapies. *PLOS ONE*, 12(10):e0186906.

[70] International Union of Physiological Sciences (2016). *About IUPS Physiome Project*.

[71] Irwin, J. et al. (1935). Tests of significance for differences between percentages based on small numbers. *Metron*, 12(2):84–94.

[72] Ivorra, J. L., Rivero, O., Costas, J., Iniesta, R., Arrojo, M., Ramos-Ríos, R., Carracedo, A., Palomo, T., Rodriguez-Jimenez, R., Cervilla, J., Gutiérrez, B., Molina, E., Arango, C., Álvarez, M., Pascual, J. C., Pérez, V., Saiz, P. A., García-Portilla, M. P., Bobes, J., González Pinto, A., Zorrilla, I. n., Haro, J. M., Bernardo, M., Baca García, E., González, J. C., Hoenicka, J., Moltó, M. D., and Sanjuán, J. (2014). Replication of previous genome-wide association studies of psychiatric diseases in a large schizophrenia case-control sample from Spain. *Schizophrenia Research*, 159(1):107–113.

[73] Jain, A., Jianchang Mao, and Mohiuddin, K. (1996). Artificial neural networks: a tutorial. *Computer (Long. Beach. Calif).*, 29(3):31–44.

[74] Jakulin, A. and Bratko, I. (2003). Analyzing attribute dependencies. In *Knowledge Discovery in Databases: PKDD 2003*, pages 229–240. Springer Berlin Heidelberg.

[75] Jiang, R., Tang, W., Wu, X., and Fu, W. (2009). A random forest approach to the detection of epistatic interactions in case-control studies. *BMC Bioinformatics*, 10(S1).

[76] Jiang, X., Jao, J., and Neapolitan, R. (2015). Learning Predictive Interactions Using Information Gain and Bayesian Network Scoring. *PLOS ONE*, 10(12):e0143247.

[77] Kantardzic, M. (2011). *Data mining: concepts, models, methods, and algorithms*. IEEE Press Wiley, Piscataway, New Jersey Hoboken, NJ.

[78] Kasper, D. L., Hauser, S. L., Jameson, J. L., Fauci, A. S., Longo, D. L., and Loscalzo, J. (2015). *Harrison's Principles of Internal Medicine*. Mc Graw Hill Education.

[79] Khan, J., Wei, J. S., Ringnér, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C. R., Peterson, C., and Meltzer, P. S. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(6):673–679.

[80] Khan, M. A., Akari, H., Kao, S., Aberham, C., Davis, D., Buckler-White, A., and Strebel, K. (2002). Intravirion processing of the human immunodeficiency virus type 1 vif protein by the viral protease may be correlated with vif function. *Journal of Virology*, 76(18):9112–9123.

[81] Kingma, D. and Ba, J. (2015). Adam: a method for stochastic optimization. In: International Conference on Learning Representations (ICLR).

[82] Kira, K. and Rendell, L. A. (1992). A practical approach to feature selection. In *Machine Learning Proceedings 1992*, pages 249–256. Elsevier.

[83] Koito, A. and Ikeda, T. (2013). Intrinsic immunity against retrotransposons by APOBEC cytidine deaminases. *Frontiers in Microbiology*, 4.

[84] Koo, C. L., Liew, M. J., Mohamad, M. S., and Salleh, A. H. M. (2013). A review for detecting gene-gene interactions using machine learning methods in genetic epidemiology. *BioMed Research International*, 2013:1–13.

[85] Korber, B. (2001). Evolutionary and immunological implications of contemporary HIV-1 variation. *British Medical Bulletin*, 58(1):19–42.

[86] Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., and Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8–17.

[87] Lapuschkin, S., Wäldchen, S., Binder, A., Montavon, G., Samek, W., and Müller, K.-R. (2019). Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications*, 10(1).

[88] Legg, S. and Hutter, M. (2007). A Collection of Definitions of Intelligence. In *Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms*, volume 157, page 17–24, Amsterdam. IOS Press.

[89] Letko, M., Booiman, T., Kootstra, N., Simon, V., and Ooms, M. (2015). Identification of the HIV-1 vif and human APOBEC3g protein interface. *Cell Reports*, 13(9):1789–1799.

[90] Lever, A. M. (2009). HIV: the virus. *Medicine*, 37(7):313–316.

[91] Li, J., Malley, J. D., Andrew, A. S., Karagas, M. R., and Moore, J. H. (2016). Detecting gene-gene interactions using a permutation-based random forest method. *BioData Mining*, 9(1):14.

[92] Li, Q., Kim, Y., Suktitipat, B., Hetmanski, J. B., Marazita, M. L., Duggal, P., Beaty, T. H., and Bailey-Wilson, J. E. (2015). Gene-Gene Interaction Among WNT Genes for Oral Cleft in Trios. *Genetic Epidemiology*, 39(5):385–394.

[93] Liu, R., Paxton, W. A., Choe, S., Ceradini, D., Martin, S. R., Horuk, R., MacDonald, M. E., Stuhlmann, H., Koup, R. A., and Landau, N. R. (1996). Homozygous defect in HIV-1 coreceptor accounts for resistance of some multiply-exposed individuals to HIV-1 infection. *Cell*, 86(3):367–377.

[94] Liu, X., Faes, L., Kale, A. U., Wagner, S. K., Fu, D. J., Bruynseels, A., Mahendiran, T., Moraes, G., Shamdas, M., Kern, C., Ledsam, J. R., Schmid, M. K., Balaskas, K., Topol, E. J., Bachmann, L. M., Keane, P. A., and Denniston, A. K. (2019). A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. *The Lancet Digital Health*, 1(6):e271–e297.

[95] López-Vázquez, G., Ornelas-Rodriguez, M., Espinal, A., Soria-Alcaraz, J. A., Rojas-Domínguez, A., Puga-Soberanes, H. J., Carpio, J. M., and Rostro-Gonzalez, H. (2019). Evolutionary spiking neural networks for solving supervised classification problems. *Computational Intelligence and Neuroscience*, 2019:1–13.

[96] Luckheeram, R. V., Zhou, R., Verma, A. D., and Xia, B. (2012). CD4+t cells: Differentiation and functions. *Clinical and Developmental Immunology*, 2012:1–12.

[97] Lustgarten, J., Balasubramanian, J., Visweswaran, S., and Gopalakrishnan, V. (2017). Learning parsimonious classification rules from gene expression data using bayesian networks with local structure. *Data*, 2(1):5.

[98] Ma, D., Whitehead, P., Menold, M., Martin, E., Ashley-Koch, A., Mei, H., Ritchie, M., DeLong, G., Abramson, R., Wright, H., Cuccaro, M., Hussman, J., Gilbert, J., and Pericak-Vance, M. (2005). Identification of significant association and gene-gene interaction of GABA receptor subunit genes in autism. *The American Journal of Human Genetics*, 77(3):377–388.

[99] Maimon, O. and Rokach, L. (2009). Introduction to knowledge discovery and data mining. In *Data Mining and Knowledge Discovery Handbook*, pages 1–15. Springer US.

[100] Maimon, O. and Rokach, L., editors (2010). *Data Mining and Knowledge Discovery Handbook*. Springer US, Boston, MA.

[101] Marinov, M. and Weeks, D. E. (2001). The complexity of linkage analysis with neural networks. *Human Heredity*, 51(3):169–176.

[102] Marques de Sá, J. P. (2001). *Pattern Recognition*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[103] Martin, W. A. and Fateman, R. J. (1971). The MACSYMA system. In *Proceedings of the second ACM symposium on Symbolic and algebraic manipulation - SYMSAC '71*. ACM Press.

[104] McDonald, J. (2014). *Handbook of Biological Statistics*. Sparky House Publishing, Baltimore, Maryland, 3rd ed. edition.

[105] McKinney, S. M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., Back, T., Chesus, M., Corrado, G. C., Darzi, A., Etemadi, M., Garcia-Vicente, F., Gilbert, F. J., Halling-Brown, M., Hassabis, D., Jansen, S., Karthikesalingam, A., Kelly, C. J., King, D., Ledsam, J. R., Melnick, D., Mostofi, H., Peng, L., Reicher, J. J., Romera-Paredes, B., Sidebottom, R., Suleyman, M., Tse, D., Young, K. C., Fauw, J. D., and Shetty, S. (2020). International evaluation of an AI system for breast cancer screening. *Nature*, 577(7788):89–94.

[106] Miller, J. H., Presnyak, V., and Smith, H. C. (2007). The dimerization domain of HIV-1 viral infectivity factor vif is required to block virion incorporation of APOBEC3g. *Retrovirology*, 4(1):81.

[107] Mitchell, T. (1997). *Machine Learning*. Mc Graw-Hill, New York.

[108] Mlcochova, P., Apolonia, L., Kluge, S. F., Sridharan, A., Kirchhoff, F., Malim, M. H., Sauter, D., and Gupta, R. K. (2015). Immune evasion activities of accessory proteins vpu, nef and vif are conserved in acute and chronic HIV-1 infection. *Virology*, 482:72–78.

[109] Moore, J. H. (2003). The ubiquitous nature of epistasis in determining susceptibility to common human diseases. *Human Heredity*, 56(1-3):73–82.

[110] Moore, J. H. (2004). The challenges of whole-genome approaches to common diseases. *JAMA: The Journal of the American Medical Association*, 291(13):1642–1643.

[111] Moore, J. H. (2010). Detecting, characterizing, and interpreting nonlinear gene–gene interactions using multifactor dimensionality reduction. In *Computational Methods for Genetics of Complex Traits*, pages 101–116. Elsevier.

[112] Moore, J. H. and Parker, J. S. (2002). Evolutionary computation in microarray data analysis. In *Methods of Microarray Data Analysis*, pages 23–35. Springer US.

[113] Morgello, S. (2016). HIV. In *Neurotropic Viral Infections*, pages 21–74. Springer International Publishing.

[114] Motsinger, A. A., Dudek, S. M., Hahn, L. W., and Ritchie, M. D. (2006). Comparison of neural network optimization approaches for studies of human genetics. In *Lecture Notes in Computer Science*, pages 103–114. Springer Berlin Heidelberg.

[115] Motsinger-Reif, A. A., Deodhar, S., Winham, S. J., and Hardison, N. E. (2010). Grammatical evolution decision trees for detecting gene-gene interactions. *BioData Mining*, 3(1):8.

[116] Motsinger-Reif, A. A., Dudek, S. M., Hahn, L. W., and Ritchie, M. D. (2008a). Comparison of approaches for machine-learning optimization of neural networks for detecting gene-gene interactions in genetic epidemiology. *Genetic Epidemiology*, 32(4):325–340.

[117] Motsinger-Reif, A. A., Lee, S. L., Mellick, G., and Ritchie, M. D. (2006). GPNN: power studies and applications of a neural network method for detecting gene-gene interactions in studies of human disease. *BMC Bioinformatics*, 7(1):39.

[118] Motsinger-Reif, A. A., Reif, D. M., Fanelli, T. J., and Ritchie, M. D. (2008b). A comparison of analytical methods for genetic association studies. *Genetic Epidemiology*, 32(8):767–78.

[119] Motsinger-Reif, A. A. and Ritchie, M. D. (2008). Neural networks for genetic epidemiology: past, present, and future. *BioData Mining*, 1(1):3.

[120] N. V. Chawla, K. W. Bowyer, L. O. H. and Kegelmeyer, W. P. (2002). Smote: Synthetic Minority Over-sampling Technique. *J. Artificial Intelligence Research*, 16:321–357.

[121] Nakashima, M., Tsuzuki, S., Awazu, H., Hamano, A., Okada, A., Ode, H., Maejima, M., Hachiya, A., Yokomaku, Y., Watanabe, N., Akari, H., and Iwatani, Y. (2017). Mapping region of human restriction factor APOBEC3h critical for interaction with HIV-1 vif. *Journal of Molecular Biology*, 429(8):1262–1276.

[122] Naranbhai, V. and Carrington, M. (2017). Host genetic variation and HIV disease: from mapping to mechanism. *Immunogenetics*, 69(8-9):489–498.

[123] National Human Genome Research Institute (2015). An overview of the Human Genome Project.

[124] National Institute of Standards and Technology (NIST) (2015). NIST big data interoperability framework: Volume 1, definitions. Technical report.

[125] Noble, W. S. (2009). How does multiple testing correction work? *Nature Biotechnology*, 27(12):1135–1137.

[126] Norris, P. J., Moffett, H. F., Yang, O. O., Daniel, E., Clark, M. J., Addo, M. M., Eric, S., Kaufmann, D. E., and Rosenberg, E. S. (2004). Beyond Help : Direct Effector Functions of Human Immunodeficiency Virus Type Beyond Help : Direct Effector Functions of Human Immunodeficiency Virus Type 1-Specific CD4+ T Cells. *Journal of Virology*, 78(16):8844–8851.

[127] Ooms, M., Letko, M., Binka, M., and Simon, V. (2013). The resistance of human APOBEC3h to HIV-1 NL4-3 molecular clone is determined by a single amino acid in vif. *PLoS ONE*, 8(2):e57744.

[128] Organization, W. H. (2016). *Consolidated guidelines on the use of antiretroviral drugs for treating and preventing HIV infection : recommendations for a public health approach.*

[129] Oriol, J. D. V., Vallejo, E. E., Estrada, K., Peña, J. G. T., and Initiative, T. A. D. N. (2019). Benchmarking machine learning models for late-onset alzheimer's disease prediction from genomic data. *BMC Bioinformatics*, 20(1).

[130] Palmer, B. E., Boritz, E., Blyveis, N., and Wilson, C. C. (2002). Discordance between frequency of human immunodeficiency virus type 1 (HIV-1)-specific gamma interferon-producing CD4(+) T cells and HIV-1-specific lymphoproliferation in HIV-1-infected subjects with active viral replication. *Journal of Virology*, 76(12):5925–5936.

[131] Pearson, K. (1901). LIII. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[132] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[133] Pilcher, C. D., Eron, J. J., Galvin, S., Gay, C., and Cohen, M. S. (2004). Acute HIV revisited: new opportunities for treatment and prevention. *Journal of Clinical Investigation*, 113(7):937–945.

[134] Pohlert, T. (2014). *The Pairwise Multiple Comparison of Mean Ranks Package (PMCMR)*. R package.

[135] Poropatich, K. and Sullivan, D. J. (2010). Human immunodeficiency virus type 1 long-term non-progressors: the viral, genetic and immunological basis for disease non-progression. *Journal of General Virology*, 92(2):247–268.

[136] Quade, D. (1979). Using weighted rankings in the analysis of complete blocks with additive block effects. *Journal of the American Statistical Association*, 74(367):680.

[137] Quinlan, J. R. (1983). Learning efficient classification procedures and their application to chess end games. In *Machine Learning*, pages 463–482. Springer Berlin Heidelberg.

[138] Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

[139] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, Inc.

[140] Ramón y Cajal, S. (1909). *Histologie du système nerveux de l'homme & des vertébrés*. Maloine, Paris.

[141] Rangel, H. R., Garzaro, D., Rodríguez, A. K., Ramírez, A. H., Ameli, G., Gutiérrez, C. D. R., and Pujol, F. H. (2009). Deletion, insertion and stop codon mutations in vif genes of HIV-1 infecting slow progressor patients. *The Journal of Infection in Developing Countries*, 3(07):531–538.

[142] Rich, E. (1983). *E. Rich. Artificial Intelligence*. McGraw–Hill, New York.

[143] Riffenburgh, R. H. (2006). Tests on categorical data. In *Statistics in Medicine*, pages 241–279. Elsevier.

[144] Ritchie, M. D., Hahn, L. W., Roodi, N., Bailey, L. R., Dupont, W. D., Parl, F. F., and Moore, J. H. (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *The American Journal of Human Genetics*, 69(1):138–147.

[145] Ritchie, M. D., White, B. C., Parker, J. S., Hahn, L. W., and Moore, J. H. (2003). Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases. *BMC Bioinformatics*, 4:28.

[146] Robinson, M. A. (1998). Linkage disequilibrium. In *Encyclopedia of Immunology*, pages 1586–1588. Elsevier.

[147] Rodger, A. J., Cambiano, V., Bruun, T., Vernazza, P., Collins, S., Degen, O., Corbelli, G. M., Estrada, V., Geretti, A. M., Beloukas, A., Raben, D., Coll, P., Antinori, A., Nwokolo, N., Rieger, A., Prins, J. M., Blaxhult, A., Weber, R., Eeden, A. V., Brockmeyer, N. H., Clarke, A., del Romero Guerrero, J., Raffi, F., Bogner, J. R., Wandeler, G., Gerstoft, J., Gutiérrez, F., Brinkman, K., Kitchen, M., Ostergaard, L., Leon, A., Ristola, M., Jessen, H., Stellbrink, H.-J., Phillips, A. N., Lundgren, J., Coll, P., Cobarsi, P., Nieto, A., Meulbroek, M., Carrillo, A., Saz, J., Guerrero, J. D., García, M. V., Gutiérrez, F., Masiá, M., Robledano, C., Leon, A., Leal, L., Redondo, E. G., Estrada, V. P., Marquez, R., Sandoval, R., Viciana, P., Espinosa, N., Lopez-Cortes, L., Podzamczer, D., Tiraboschi, J., Morenilla, S., Antela, A., Losada, E., Nwokolo, N., Sewell, J., Clarke, A., Kirk, S., Knott, A., Rodger, A. J., Fernandez, T., Gompels, M., Jennings, L., Ward, L., Fox, J., Lwanga, J., Lee, M., Gilson, R., Leen, C., Morris, S., Clutterbuck, D., Brady, M., Asboe, D., Fedele, S., Fidler, S., Brockmeyer, N., Potthoff, A., Skaletz-Rorowski, A., Bogner, J., Seybold, U., Roider, J., Jessen, H., Jessen, A., Ruzicic, S., Stellbrink, H.-J., Kümmerle, T., Lehmann, C., Degen, O., Bartel, S., Hüfner, A., Rockstroh, J., Mohrmann, K., Boesecke, C., Krznaric, I., Ingiliz, P., Weber, R., Grube, C., Braun, D., Günthard, H., Wandeler, G., Furrer, H., Rauch, A., Vernazza, P., Schmid, P., Rasi, M., Borso, D., Stratmann, M., Caviezel, O., Stoeckle, M., Battegay, M., Tarr, P., Christinet, V., Jouinot, F., Isambert, C., Bernasconi, E., Bernasconi, B., Gerstoft, J., Jensen, L. P., Bayer, A. A., Ostergaard, L., Yehdego, Y., Bach, A., Handberg, P., Kronborg, G., s. Pedersen, S., Bülow, N., Ramskover, B., Ristola, M., Debnam, O., Sutinen, J., Blaxhult, A., Ask, R., Hildingsson-Lundh, B., Westling, K., Frisen, E.-M., Cortney, G., O'Dea, S., Wit, S. D., Necsoi, C., Vandekerckhove, L., Goffard, J.-C., Henrard, S., Prins, J., Nobel, H.-H., Weijsenfeld, A., Eeden, A. V., Elsenburg, L., Brinkman, K., Vos, D., Hoijenga, I., Gisolf, E., Bentum, P. V., Verhagen, D., Raffi, F., Billaud, E., Ohayon, M., Gosset, D., Fior, A., Pialoux, G., Thibaut, P., Chas, J., Leclercq, V., Pechenot, V., Coquelin, V., Pradier, C., Breaud, S., Touzeau-Romer, V., Rieger, A., Geit, M. K. M., Sarcletti, M., Gisinger, M., Oellinger, A., Antinori, A., Menichetti, S., Bini, T., Mussini, C., Meschiari, M., Biagio, A. D., Taramasso, L., Celesia, B. M., Gussio, M., and Janeiro, N. (2019). Risk of HIV transmission through condomless sex in serodifferent gay couples with the HIV-positive partner taking suppressive antiretroviral therapy (PARTNER): final results of a multicentre, prospective, observational study. *The Lancet*, 393(10189):2428–2438.

[148] Rodríguez-Escobedo, Gilberto, J., García-Sepúlveda, C. A., and Cuevas-Tello, J. C. (2015). KIR Genes and Patterns Given by the A Priori Algorithm: Immunity for Haematological Malignancies. *Computational and Mathematical Methods in Medicine*, 2015:1–11.

[149] Rokach, L. and Maimon, O. (2009). Classification Trees. In *Data Mining Knowledge Discovery Handbook*, pages 149–174. Springer US, Boston, MA.

[150] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408.

[151] Rowland-Jones, S. L. and Whittle, H. C. (2007). Out of africa: what can we learn from HIV-2 about protective immunity to HIV-1? *Nature Immunology*, 8(4):329–331.

[152] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

[153] Russell, S. and Norvig, P. (2010). *Artificial intelligence : a modern approach*. Prentice Hall, Upper Saddle River, New Jersey.

[154] Salgado, M., Kwon, M., Gálvez, C., Badiola, J., Nijhuis, M., Bandera, A., Balsalobre, P., Miralles, P., Buño, I., Martinez-Laperche, C., Vilaplana, C., Jurado, M., Clotet, B., Wensing, A., Martinez-Picado, J., and and, J. L. D.-M. (2018). Mechanisms that contribute to a profound reduction of the HIV-1 reservoir after allogeneic stem cell transplant. *Annals of Internal Medicine*, 169(10):674.

[155] Santhiveeran, J. (2006). Data mining for web site evaluation. *Journal of Teaching in Social Work*, 26(3-4):181–196.

[156] Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423.

[157] Sheehy, A. M., Gaddis, N. C., Choi, J. D., and Malim, M. H. (2002). Isolation of a human gene that inhibits HIV-1 infection and is suppressed by the viral vif protein. *Nature*, 418(6898):646–650.

[158] Shepherd, G. M., editor (2004). *The Synaptic Organization of the Brain*. Oxford University Press.

[159] Shmilovici, A. (2009). Support Vector Machines. In *Data Mining Knowledge Discovery Handbook*, pages 231–247. Springer US, Boston, MA.

[160] Shortliffe, E., editor (1976). *Computer-Based Medical Consultations: Mycin*. Elsevier.

[161] Smith, J. L., Bu, W., Burdick, R. C., and Pathak, V. K. (2009). Multiple ways of targeting APOBEC3–virion infectivity factor interactions for anti-HIV-1 drug development. *Trends in Pharmacological Sciences*, 30(12):638–646.

[162] Smyth, R. P., Davenport, M. P., and Mak, J. (2012). The origin of genetic diversity in HIV-1. *Virus Research*, 169(2):415–429.

[163] Soros, V. B. and Greene, W. C. (2007). APOBEC3g and HIV-1: Strike and counterstrike. *Current HIV/AIDS Reports*, 4(1):3–9.

[164] Steele, B., Chandler, J., and Reddy, S. (2016). *Algorithms for Data Science*. Springer International Publishing.

[165] Strebel, K. (2013). HIV accessory proteins versus host restriction factors. *Current Opinion in Virology*, 3(6):692–699.

[166] Tan, P.-N., Steinbach, M., and Kumar, V. (2013). *Introduction to Data Mining*.

[167] Tenesa, A. and Haley, C. S. (2013). The heritability of human disease: estimation, uses and abuses. *Nature Reviews Genetics*, 14(2):139–149.

[168] Theodoridis, S. (2009). *Pattern recognition*. Academic Press, Burlington, MA London.

[169] Tong, D. L., Boocock, D. J., Dhondalay, G. K. R., Lemetre, C., and Ball, G. R. (2014). Artificial Neural Network Inference (ANNI): A study on gene-gene interaction for biomarkers in childhood sarcomas. *PLoS One*, 9(7):1–13.

[170] Tong, D. L. and Schierz, A. C. (2011). Hybrid genetic algorithm-neural network: Feature extraction for unpreprocessed microarray data. *Artificial Intelligence in Medicine*, 53(1):47–56.

[171] Topalovic, M., Das, N., Burgel, P.-R., Daenen, M., Derom, E., Haenebalcke, C., Janssen, R., Kerstjens, H. A., Liistro, G., Louis, R., Ninane, V., Pison, C., Schlesser, M., Vercauter, P., Vogelmeier, C. F., Wouters, E., Wynants, J., and Janssens, W. (2019). Artificial intelligence outperforms pulmonologists in the interpretation of pulmonary function tests. *European Respiratory Journal*, 53(4):1801660.

[172] Turing, A. M. (1950). I.—Computing Machinery and Intelligence. *Mind*, LIX(236):433–460.

[173] Turk, G., Ghiglione, Y., Hormanstorfer, M., Laufer, N., Coloccini, R., Salido, J., Trifone, C., Ruiz, M., Falivene, J., Holgado, M., Caruso, M., Figueroa, M., Salomón, H., Giavedoni, L., Pando, M., Gherardi, M., Rabinovich, R., Pury, P., and Sued, O. (2018). Biomarkers of progression after HIV acute/early infection: Nothing compares to CD4+ t-cell count? *Viruses*, 10(1):34.

[174] UNAIDS (2018). UNAIDS data 2018. http://www.unaids.org/en/resources/documents/2018/unaids-data-2018. Accessed: 2018-07-09.

[175] UNAIDS (2019). UNAIDS Fact sheet - Worlds AIDS day 2019. https://www.unaids.org/sites/default/files/media_asset/UNAIDS_FactSheet_en.pdf. Accessed: 2020-01-23.

[176] U.S. Census Bureau (2018). Comparative Demographic Estimates, 2018 American Community Survey 1-Year Estimates, Table: CP05. http://data.census.gov. Accessed: 2020-06-05.

[177] Weber, I. T. and Harrison, R. W. (2017). Decoding HIV resistance: from genotype to therapy. *Future Medicinal Chemistry*, 9(13):1529–1538.

[178] Weiss, R. (1993). How does HIV cause AIDS? *Science*, 260(5112):1273–1279.

[179] Witten, I. H., Frank, E., and Hall, M. A. (2017). *Data mining : Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Cambridge, MA.

[180] Yang, S., Sun, Y., and Zhang, H. (2000). The multimerization of human immunodeficiency virus type i vif protein. *Journal of Biological Chemistry*, 276(7):4889–4893.

[181] Yu, Y., Xiao, Z., Ehrlich1, E. S., Yu, X., , and Yu, X.-F. (2004). Selective assembly of HIV-1 vif-cul5-ElonginB-ElonginC e3 ubiquitin ligase complex through a novel SOCS box and upstream cysteines. *Genes & Development*, 18(23):2867–2872.

[182] Zhang, W., Chen, G., Niewiadomska, A. M., Xu, R., and Yu, X.-F. (2008). Distinct determinants in HIV-1 vif and human APOBEC3 proteins are required for the suppression of diverse host anti-viral proteins. *PLoS ONE*, 3(12):e3963.

[183] Zhang, Z., , Beck, M. W., Winkler, D. A., Huang, B., Sibanda, W., and Goyal, H. (2018). Opening the black box of neural networks: methods for interpreting neural network models in clinical applications. *Annals of Translational Medicine*, 6(11):216–216.