Universidade do Minho
Escola de Engenharia
Departamento de Informática

Joana dos Santos Martins

HLA Binding Intelligence (HABIT)

An integrated web-server for generation and advanced interpretation of peptide-HLA binding predictions

October 2017

Universidade do Minho
Escola de Engenharia
Departamento de Informática

Joana dos Santos Martins

HLA Binding Intelligence (HABIT)

An integrated web-server for generation and advanced interpretation of peptide-HLA binding predictions

Master dissertation
Master Degree in Bioinformatics

Dissertation supervised by
Nuno Miguel Sampaio Osório
Miguel Francisco Almeida Pereira Rocha

October 2017

ACKNOWLEDGEMENTS

No momento de concluir a presente tese, gostaria de agradecer a todas as pessoas que direta ou indiretamente me acompanharam e ajudaram ao longo desta etapa. Desta forma, deixo aqui expresso um especial agradecimento:

Ao Nuno Osório, pela oportunidade de desenvolver este projeto, pelos constantes desafios lançados, pela orientação e apoio incondicional que muito contribuíram para o meu conhecimento científico. Mas principalmente, um enorme obrigada pela confiança e liberdade depositada em mim desde o início, sempre acompanhadas por palavras de incentivo e doses de otimismo que foram essenciais na minha vontade de fazer sempre mais e melhor.

Ao professor Miguel Rocha, com quem muito aprendi nestes últimos dois anos. Quero agradecer pela constante disponibilidade, pelas recomendações e ideias, pela dedicação e pelo rigor científico incutidos em todas as fases do projeto.

Ao Carlos e ao Pedro, pela ajuda e conselhos dados, pelos variados *brainstormings*, pelas discussões científicas e pelas múltiplas questões de programação que resolvemos. Obrigada pelo ambiente de trabalho que proporcionaram, pela amizade e boa disposição.

Aos meus queridos amigos, por estarem presentes em todos os momentos.

Aos meus pais e aos meus irmãos, pelo apoio em todas as minhas decisões e pelo amor incondicional, o meu maior obrigada.

A todos, muito obrigada!

ABSTRACT

Human T cells are essential in the control of pathogen infections, cancer and autoimmune diseases. Immune responses mediated by T cells are aimed at specific peptides, designated T cell epitopes, that are recognized when bound to *Human leukocyte antigen (HLA)* molecules. The HLA genes are remarkably polymorphic in the human population driving a broad and fine-tuned capacity of their encoded proteins to bind a wide array of peptide sequences. Amino acid variants in epitopes might impact the HLA-peptide interaction and consequently the level and type of generated T cell responses. Having the tools to effectively predict and measure the impact of amino acid variants in HLA binding will be of great value for the future of personalized host directed therapies.

Multiple algorithms based on *Machine learning (ML)* have been implemented to estimate HLA-peptide binding. However, there is still no tool capable of performing integrated analyses, namely comparing wild type and mutant sequences by predicting all overlapping peptides including amino acid positions of interest. This requires that researchers have programming skills to analyse prediction data for HLA peptide binding and to extract potentially meaningful conclusions from that data.

The main objective of this thesis was to design and implement a web server, called *HLA Binding Intelligence (HABIT)*, that automates HLA binding prediction and advanced interpretation of the impact of peptide variants in inducing adaptive immune responses. HABIT integrates the best overall predictors for peptide-HLA binding (class I and II), *NetMHCpan* and *NetMHCIIpan*, respectively. The application features an intuitive and user-friendly interface available online for academic users, including comparative studies of wild type and mutant sequences, statistical tests and calculations of human population coverage. The performance of the web server was tested with a case study representative of tuberculosis, the R233L mutation found in the *acetyl-/propionyl-CoA carboxylase beta chain (AccD2)* protein. The results obtained through the interactive graphical interface allowed the automated identification of the most promising peptide sequences to be used in the design of T cell-mediated immunotherapy approaches.

Overall, HABIT provides a quick and easy way to answer major questions of rational immunotherapy studies, namely “What is the influence of an amino acid variants on HLA binding?”, “Do wild type and mutant epitopes show statistically significant differences in predicted HLA binding properties?” and “What is the predicted population coverage of the epitopes?”. Thus, HABIT constitutes a promising and reliable advance in the discovery of molecular determinants that influence the variation in T cell mediated immune responses.

RESUMO

No ser humano, as células T são essenciais no controle de infeções patogénicas, do cancro e das doenças autoimunes. As respostas imunológicas mediadas por células T destinam-se a péptidos específicos, designados por epítopos de células T, que são reconhecidos quando ligados previamente a moléculas de HLA. Os genes de HLA são extremamente polimórficos na população humana, deste modo as proteínas que codificam, apresentam uma ampla e aperfeiçoada capacidade para se ligarem a um enorme conjunto de sequências peptídicas. As variantes de aminoácidos encontradas nos epítopos podem afetar a interação HLA-péptido e consequentemente o nível e tipo de respostas geradas pelas células T. A existência de ferramentas capazes de prever e de medir o impacto destas variantes de aminoácidos na ligação das moléculas de HLA será de grande valor para o futuro das terapias personalizadas.

Múltiplos algoritmos baseados em *machine learning* foram implementados para estimar a ligação do péptido ao HLA. No entanto, ainda não existe uma ferramenta capaz de realizar análises integradas, nomeadamente comparar sequências do tipo selvagem e mutante através da previsão de todos os péptidos que incluem a posição do aminoácido de interesse. Isso exige que os investigadores tenham conhecimentos de programação para analisar dados de previsão para a ligação do péptido ao HLA e extrair conclusões potencialmente significativas desses dados.

A presente tese teve como principal objetivo projetar e implementar uma aplicação, designada de HABIT, que automatiza o processo de previsão da ligação ao HLA e a interpretação avançada do impacto de variantes peptídicas na indução de respostas imunes adaptativas. O HABIT integra as melhores ferramentas de previsão geral para a ligação dos péptidos às moléculas HLA (classe I e II), o *NetMHCpan* e o *NetMHCIIpan*, respetivamente. O servidor apresenta uma interface intuitiva e *user-friendly* disponível online para os utilizadores académicos, incluindo estudos comparativos das sequências selvagem e mutante, testes estatísticos e cálculos de cobertura de população humana. O desempenho do servidor web foi testado com um caso de estudo representativo da tuberculose, a mutação R233L encontrada na proteína AccD2. Os resultados obtidos através da interface gráfica interativa permitiram de forma automatizada identificar as sequências de péptidos mais promissoras a serem utilizadas em projetos no contexto da imunoterapia.

Em geral, o HABIT fornece uma maneira rápida e fácil de responder às principais questões de estudo de imunoterapia racional, nomeadamente "Qual é a influência de uma variante de aminoácido na ligação aos alelos?", "Os epítopos selvagem e mutante apresentam diferenças estatisticamente significativas nas propriedades de ligação ao HLA?" e "Qual é a cobertura populacional prevista para os epítopos?". Assim, o HABIT constitui um avanço promissor e confiável na descoberta de determinantes moleculares que influenciam a variação nas respostas imunes mediadas por células T.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

A

AccD2 acetyl-/propionyl-CoA carboxylase beta chain.

ANNs Artificial Neural Networks.

B

BCG bacillus Calmette-Gurin.

C

$CD4^+$ Cluster of differentiation 4 positive.

$CD8^+$ Cluster of differentiation 8 positive.

H

HABIT HLA Binding Intelligence.

HLA Human leukocyte antigen.

I

IEDB Immune Epitope Database and Analysis Resource.

M

MHC Major histocompatibility complexes.

ML Machine learning.

P

PSSM Position-specific scoring matrix.

S

SB Strong binder.

T

TB Tuberculosis.

W

WB Weak binder.

WHO World Health Organization.

1

INTRODUCTION

1.1 CONTEXT AND MOTIVATION

The main role of the immune system is to recognize and defend the body against invading pathogenic organisms, and also against damaged biomolecules and dysfunctional cells. The removal of these threats is essential to prevent the development of infectious, oncologic and autoimmune diseases (Caspi, 2008).

The immune system can be divided into two main branches, the innate and the adaptive. The innate immune system is a less specific, but faster mechanism of response to threat patterns, such as pathogen-associated molecular patterns and damage-associated molecular patterns. It can be considered a first line of defence, activated by several pattern recognition receptors, including toll-like receptors, and its action is mediated by several immune cells, such as macrophages/monocytes, neutrophils, natural killer cells, basophils, and eosinophils (Medzhitov and Jr, 2000).

In contrast, the adaptive immune system is mainly triggered by the recognition of specific amino acid sequences, epitopes, in peptidic antigens that are presented to T and B lymphocytes, the major effectors of adaptive immune responses (Kanaly et al., 2010; Parkin and Cohen, 2001). An epitope is the part of an antigen that is recognized by B or T cell receptors. T-cell receptors only recognize epitopes that are presented in the context of *Major histocompatibility complexes (MHC)*. In humans, the MHC is also called the HLA, and these proteins are divided into HLA class I and II (Rosa et al., 2010).

Accurate prediction of the peptides that can be epitopes is a crucial step for immunotherapy, which shows very promising results in the treatment of different diseases, such as cancer (Parsons et al., 2008). A potential immunotherapy target are cancer neoantigens (Schumacher and Schreiber, 2015). These are generated by tumor-specific amino acid variants with impact in the HLA-peptide interaction and consequently the level and type of generated T cell responses (Kreiter et al., 2015). Prediction of these epitopes consists in selecting the protein sequence and estimating different parameters of HLA binding affinity and stability of each peptide fragment, allowing peptides with strongest binding prediction to be chosen (Oyarzun and Kobe, 2015).

Experimental HLA-binding assays are very expensive and time-consuming, so one of the main goals of immunoinformatics is to develop efficient algorithms trained on biological data, which are capable of predicting HLA binding. The software tools *NetMHCpan* 3.0 (Nielsen and Andreatta, 2016) and *NetMHCIIpan* 3.1 (Andreatta et al., 2015) implement ML approaches to estimate peptide-HLA binding and are considered the best overall predictors for class I (Trolle et al., 2015) and class II (Zhang et al., 2012b) HLAs, respectively.

In addition to finding peptides that strongly bind to class I and II HLAs, there are other main challenges to design effective vaccination strategies. On the one hand, individuals harbour a diversified set of HLA alleles, which often underlie individualized and differential responses to specific peptides. On the other hand, HLA alleles are expressed at different levels in different tissues, cells or in response to different stimuli (Oyarzun and Kobe, 2015).

Presently, several tools are available for the *in silico* HLA-binding prediction stage, but these tools run individual analysis and do not perform comparison of the results impairing the evaluation of the impact of amino acid variants in the peptide-HLA binding properties (Farrell and Gordon, 2015). Specifically, this type of analysis requires individual prediction of wild type and mutant sequences, the selection of all overlapping peptides including amino acid positions of interest and the comparison of the results that were generated separately.

Since other tools do not execute these comparative tasks, programming skills are needed to extract meaningful conclusions from large outputs of HLA-peptide binding prediction data, thus delaying the research advances in this relevant topic. Furthermore, additional data prioritization strategies such as statistical tests or population coverage calculation are inexistent in an integrated manner and would be of great value for the rational design of T cell-based vaccination strategies (Oyarzun and Kobe, 2015).

1.2 OBJECTIVES

The main goal of this project is to develop a web server, named HLA Binding Intelligence (HABIT), that provides an integrated analysis system with a user friendly interface, and shows the predicted influence of peptide variants in immune response, through a set of summary graphs and tables, that can be explored online or exported to be used in publications.

In detail, the scientific and technological objectives of this work are:

1. Develop a set of bash scripts to automate the export and processing of data from the epitopes prediction tools.

2. Filter and organize the most important results of each submitted project.

3. Design a web application that shows the influence of peptides on the immune response, implemented through the R scientific computation systems, with the *shiny* package.

Accomplishing these objectives will allow the:

- Generation of mutant protein sequences from wild type protein input files (FASTA format) with user-defined amino acid substitutions;
- Integration of epitope prediction tools into the web server;
- Simultaneous HLA-binding prediction with *NetMHCpan* and *NetMHCIIpan* for wild type and mutant sequences with no limits imposed to the number of peptides, their lengths and HLA molecules, which enables high-throughput approaches;
- Calculation of the differences between wild type and mutant sequences (including percentage of predicted epitopes, mean HLA affinity and mean percentage rank), including the analysis of statistical significance;
- Information about HLA distribution across the globe in the results, as a way to evaluate the predicted impact of epitope variants in the immune responses of the human populations;
- Interactive graphic visualization of the results, with sort and search functionalities.

1.3 STRUCTURE OF THE THESIS

This document has been divided into five separate chapters, which address the following topics:

Chapter 1: Introduction

The current chapter aimed to make a general introduction to the thesis, including the context and motivation, the proposed objectives and the structure of the document.

Chapter 2: State of the Art

This chapter included a detailed description of the state of the art of the scientific and technological fields related to the thesis, namely the functioning of the immune system, the importance of immunotherapy, the advent of immunoinformatics and current strategies for vaccine design.

Chapter 3: Web server development

The process of implementing the web server and methods developed were explained in detail in this chapter. The integration of epitope prediction tools was also described, including all necessary technical concepts.

Chapter 4: Results and discussion

The results obtained through the HABIT, for a specific case study, were herein analysed and discussed. This chapter included a critical evaluation and validation of the functioning and consistency of the web server.

Chapter 5: Conclusions

The last chapter explored the general conclusions of this work, as well as its limitations and perspectives of future work.

2

STATE OF THE ART

This chapter is dedicated to explore the most relevant scientific and technical aspects related to this project. For this purpose, the immune system was characterized with a focus on its constitution and functioning. Inherently, the HLA molecules were highlighted since the understanding of its role is essential for the contextualization of this thesis. The concepts of immunotherapy and immunoinformatics, as well as their increasing importance and applications, were also addressed. This chapter culminates with the analysis and description of the most relevant existing epitope prediction tools, including *NetMHCpan* and *NetMHCIIpan*.

2.1 IMMUNE SYSTEM

The immune system is a pervasive network of molecular mediators, cells, tissues and organs with specialized roles in protecting the human body against invading pathogenic organisms and also against damaged biomolecules and dysfunctional cells (Delves and Roitt, 2000). Through a series of evolved defence mechanisms, this system recognizes and destroys threats to the organism. The success of this system is essential to prevent the development of infectious, oncologic and autoimmune diseases (Caspi, 2008). The immune system is constituted by an interplay of the two distinct branches, the innate and the adaptive (Janeway et al., 1997; Mogensen, 2009).

The innate immune system can be considered the first line of defence functioning through a fast mechanism of response to patterns, such as pathogen-associated molecular patterns (PAMPs) and damage-associated molecular patterns (DAMPs) (Alberts et al., 2003). The innate immune responses are mediated by several immune cells, such as neutrophils, macrophages, basophils, eosinophils and natural killer cells (Delves and Roitt, 2000). All these cell types recognize certain classes of conserved microbial structures, the PAMPs, thus distinguishing between self and non-self. Binding to these structures is mediated by germline-encoded pattern recognition receptors (PRRs), which are divided in different classes including Toll-like receptors (TLRs), C-type lectin receptors (CLRs), NOD-like receptors (NLRs), scavenger receptors and the RIG-like helicases (RLRs). PRRs also recognize

DAMPs that are present in endogenous molecules released from damaged cells (Basset et al., 2003; Takeuchi and Akira, 2010).

On the other hand, the adaptive immune system, also known as the acquired immune system, presents a set of receptors that trigger a response capable of recognizing and eliminating a specific pathogen or threat. This subsystem, which complements the processes of the innate immune system, creates immunological memory after a first response to a specific threat allowing future responses to be faster and more effective. This process of acquired immunity is the basis of vaccination (Alberts et al., 2003).

The major effectors of the adaptive immune system are T and B lymphocytes (Kanaly et al., 2010). B cells are the agents of humoral immunity and secrete immunoglobulins, the antibodies, known to have a relevant role in eliminating some extracellular microorganisms. The activation of B cells is mainly triggered by the recognition of an antigen, usually in the presence of helper T cells (Cooper, 2015).

T cells are responsible to mediate immune response and are divided into several groups, highlighting $CD8^+$ cytotoxic and $CD4^+$ helper cells (Delves and Roitt, 2000). B and T cells recognize antigens differently, B cells recognize intact antigens, whereas T cells recognize fragments of an antigenic protein that have been partly degraded. Particularly, an epitope, also known as antigenic determinant, is the part of an antigen that is recognized by B cells (B-cell epitopes) or T cells (T-cell epitopes). B-cell receptors recognize epitopes on the surface of protein antigens and T-cell receptors only recognize epitopes that are associated with major histocompatibility complexes (MHC) (Rosa et al., 2010; Kumar and Chandra, 2014). Overall, the recognition of epitopes presented by MHC molecules is a critical step in T cell activation and, consequently, in the development of adaptive immunity against pathogens.

2.2 MHC MOLECULES

Historically, Peter A. Gorer did in 1936 the first descriptions of the MHC (Klein, 1986) in a seminal study of tumor transplantation in mice. However, the term MHC had already been used in 1948 by George Snell (Snell, 1948). Two decades later, Zinkernagel and Doherty (1974) also published the role of MHC in a ground-breaking work on T-cell restriction. The first human MHC antigen was discovered by Jean Dausset in 1958 and termed MAC or human transplant antigen "Hu-1" (Dausset, 1958). In humans, the MHC or HLA, constitutes the most polymorphic group of genes in the whole genome, resulting in the occurrence in the population of a vast multiplicity of variants of each HLA gene (Little and Parham, 1999). The increasing complexity in genetic diversity found in the HLA system implied the establishment of rules to be applied in the systematic nomenclature of these genes. The nomination process is carried out by the Nomenclature Committee for Factors of the HLA

System of the WHO, which met for the first time in 1968. This committee meets regularly to establish and discuss issues related to nomenclature, curation and maintenance of HLA alleles.

The current WHO Nomenclature Committee is described in detail in Marsh et al. (2010). HLA molecules are divided in class I and class II molecules, which are currently constituted by 7678 alleles and 2268 alleles, respectively. The implemented nomenclature system consists of a unique code corresponding to the name of each HLA, being able to have up to four groups of numbers separated by colons (**Figure 1**). Each allele, at a minimum, presents the first two sets of numbers, depending on its sequence and the nearest relative. Overall, the HLA-A portion indicates that it is a human MHC gene and belongs to the A locus. The first two numbers (HLA-A*24) represent the serological antigen group of that allele. The HLA-A*24:02 set is assigned in sequential order of discovery. Any allele consisting solely of the described nomenclature produces a unique protein which is characteristic to it, whereby when an allele with a different DNA sequence produces a repeated protein, the third set of numbers is introduced. Thus, this set is responsible for indicating silent or non-coding substitutions. Finally, this nomenclature can be supplemented by the differences found in the non-coding regions and with reference to the level of expression.

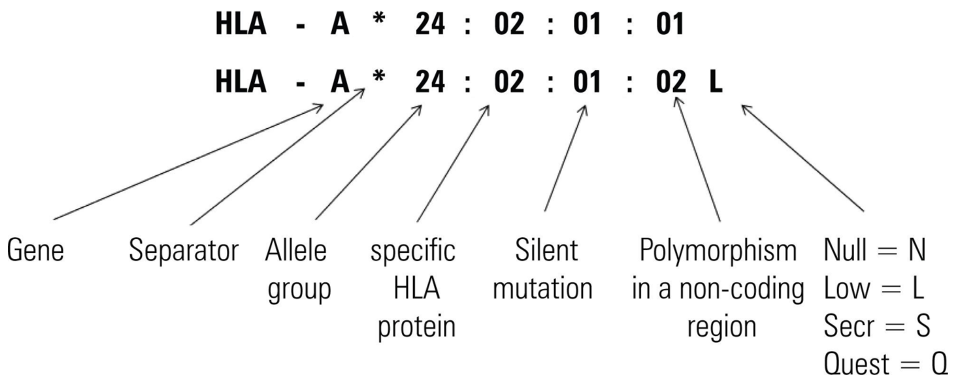


Figure 1: HLA nomenclature. Data source: Nomenclature Committee for Factors of the HLA system, 2010

The HLA complex consists of a region with a high diversity rate composed of genes related to the immune system, namely recognition of antigenic epitopes which confers the ability to fight, for example, pathogenic infections and autoimmune diseases. The association with a vast array of diseases reinforces the importance of the study and understanding of the HLA complex. This region comprises about 220 genes that are in the short arm of chromosome 6 (Bahram, 1999; Mungall et al., 2003). The HLA class I region consists of 18 genes, highlighting the classical genes (HLA-A, HLA-B and HLA-C) and non-classical genes (HLA-E and HLA-G). The molecules encoded by these genes are expressed in most cells of the human body. Meanwhile, the HLA class II region is composed of 19 genes, namely HLA-DR, HLA-DP and HLA-DQ, whose molecules are expressed mainly in antigen presenting cells (Dausset, 1981; Cano et al., 2007)

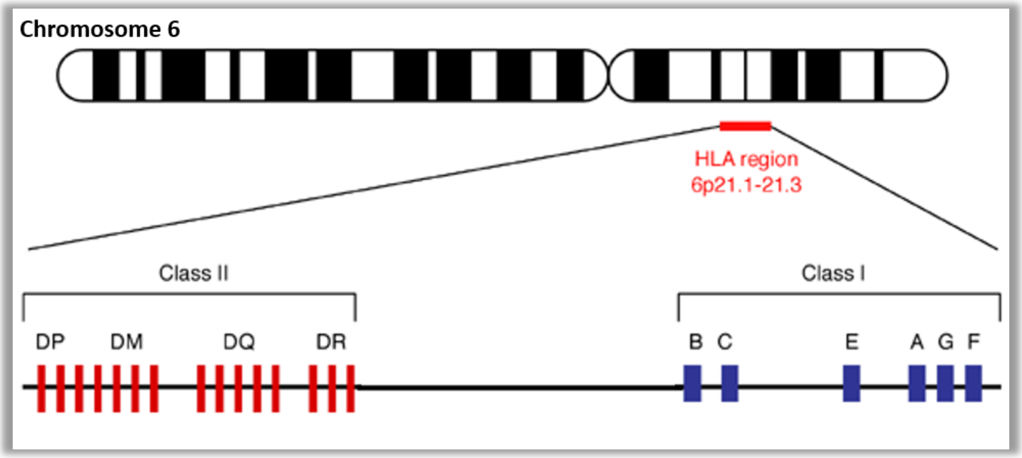


Figure 2: HLA complex. This group of genes belongs to human chromosome 6 and is responsible for encoding antigen-presenting proteins on the cell surface. HLA class I and HLA class II are in the short arm of the chromosome.

The two classes of HLA present significant differences at the structural level, as well as in the role they play in the immune system. HLA class I molecules are essentially responsible for preventing pathogenic infections and cancer by presenting intracellular peptides on the cell surface to $CD8^+$ T cells. On the other hand, class II molecules play an important role in the initiation of the immune response by binding antigenic peptides mainly acquired by phagocytosis and displayed on the cell surface for recognition by $CD4^+$ T cells (Germain, 1994), as shown in **Figure 3**. HLA class I ligands are typically 8-14 amino acids in length, while the majority of HLA class II ligands are longer, typically 9-30 amino acids in length.

HLA class I molecules are heterodimers with two polypeptide chains, the heavy chain consists in three globular domains ($\alpha1$, $\alpha2$, and $\alpha3$) and, the light chain or $\beta2$-microglobulin (**Figure 4 A**). $\alpha3$ domain and $\beta2$-microglobulin interact to form a noncovalent bond between the two chains. The $\alpha3$ domain is also responsible to interact with $CD8^+$ receptor of T cells and $\alpha1$ and $\alpha2$ domains form a groove allowing the binding of peptides.

In contrast, HLA class II molecules are expressed as heterodimers consisting of two homogenous peptides, an α and a β chain. Both chains are produced by HLA genes and have two extracellular domains, $\alpha1$ and $\alpha2$, and $\beta1$ and $\beta2$, respectively, as shown in **Figure 4 B**. The domains $\alpha1$ and $\beta1$ constitute the antigenic peptide binding site of class II HLA molecules (Rosa et al., 2010; van den Elsen et al., 1998).

Accurate prediction of the peptides that can bind to specific HLA molecules will constitute a fundamental step for the understanding and successful modulation of immune

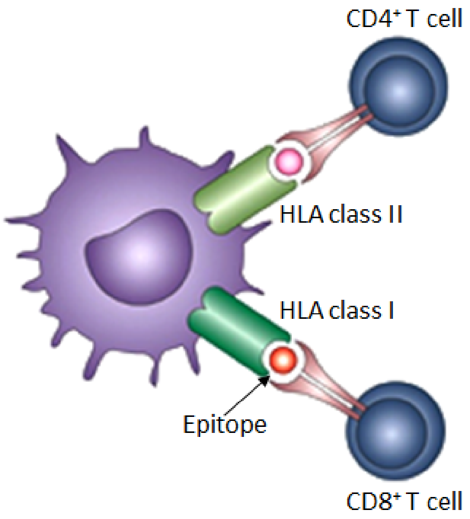


Figure 3: Immune response mediated by T cells. HLA class I and HLA class II bind to epitopes for displaying them on the cell surface for recognition by CD8$^+$ and CD4$^+$ T cells, respectively.

responses and for the discovery of novel immunogenic epitopes (Zhang et al., 2012b). Moreover, these peptides offer promising immunotherapies, which show promising results in the treatment of different human diseases (Trowsdale and Knight, 2013), such as cancer (Voutsas et al., 2007) and autoimmunity (Kong et al., 2007).

2.3 IMMUNOINFORMATICS

The search for the motifs responsible for the binding of peptides to HLA molecules started in the 1990s (Falk et al., 1991). However, the discovered complexity of HLA-peptide interactions prompted the development for this task of pattern recognition methods, namely machine learning approaches.

Machine learning is a fundamental branch of Artificial Intelligence, which intends to "learn" (or train) models from data and use these trained models to classify new data, using statistical, probabilistic and optimization methods (Mitchell, 1997). Thus, this approach can be used to analyse, interpret and predict data, returning results that would not be obtained through conventional statistics. Despite of its immense potential the success in the use of ML requires a good understanding of the limitations of the data and the specific algorithms used in each approach (Duda et al., 2001). ML has several applications in medicine, including cancer research by the use of models such as *Artificial Neural Networks (ANNs)* (Maclin et al., 1991) or decision trees (DTs) (Simes, 1985).

ANNs are one of the main ML models, providing a computational approach that loosely resembles the brain. This architecture is composed of a set of artificial neurons distributed by layers. The input layer receiving the initial data, generally normalized; the hidden or

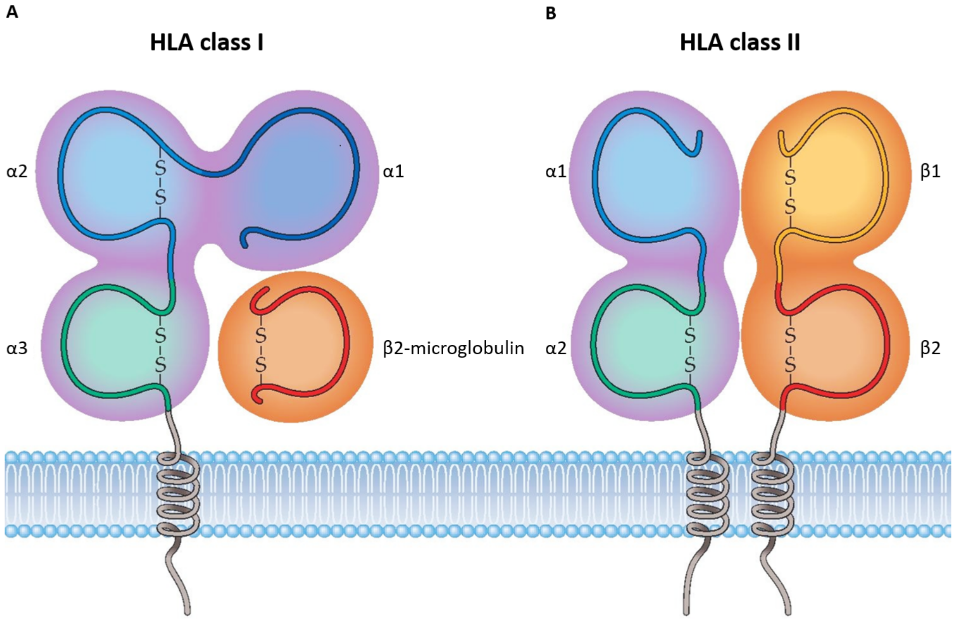


Figure 4: HLA class I and class II molecules present significant differences at the structural level. **A.** HLA class I molecules are heterodimers with two polypeptide chains, the heavy chain consists in three globular domains ($\alpha1$, $\alpha2$, and $\alpha3$) and, the light chain or $\beta2$-microglobulin. **B.** HLA class II molecules are heterodimers of α and β chains with a very similar overall structure. Both chains have two extracellular domains, $\alpha1$ and $\alpha2$, and $\beta1$ and $\beta2$, respectively.

intermediate layer that is responsible for extracting the patterns associated with the data, and the output layer which presents the results of the processing. Learning capability is one of the most important characteristics of neural networks. Thus, from a sample, the neural network learns the relationship between the input and output layers, and can produce solutions for any other example. The learning process consists of an algorithm that changes the synaptic weights, associated with the neurons, of the network iteratively to achieve a certain goal (Mitchell, 1997). The data set is usually divided into two subsets, training and testing. The training subset is used in the learning process and the test subset is used to validate a given topology.

McCulloch and Pitts (1943) developed the first model based in threshold logic, and Rumelhart et al. (1986) published the backpropagation learning rule for multilayered feedforward neural networks, a classical ANNs training algorithm, which has been complemented by a number of faster and more robust algorithms in recent years. ANNs can be used to solve problems of classification or pattern recognition, using a range of statistical and logical operations (Mitchell, 1997).

Immunoinformatics is an area of bioinformatics that focuses on the *in silico* analysis and modelling of immunological data and problems. Immunoinformatics applications are growing and, consequently, are becoming more important to immunological research (Brusic and

Petrovsky, 2003). Experimental HLA-binding assays require synthesis and testing of overlapping peptides, including the full length of the sequence of interest, which on a large scale is expensive and time-consuming. In order to lower the cost and the time needed, the primary objective of immunoinformatics research is to design efficient algorithms for mapping potential B-cell and T-cell epitopes. These tools are capable to determine the sequence regions with potential binding sites, which in turn allows accelerates the development of novel immunotherapies (Patronov and Doytchinova, 2013).

2.4 EPITOPE VACCINE DESIGN

Over the years, development of effective vaccines has been essential in the prevention and treatment of various infectious diseases (Perrie et al., 2007). An effective vaccine is characterized by the induction of a strong and long-lasting humoral and cell-mediated immune response, capable of protecting the human body against a particular disease (Li et al., 2014).

Edward Jenner, in 1796, used the term vaccination for first time to describe the new procedure of the injection of smallpox vaccine (Riedel, 2005). Later, the concept was explored by Louis Pasteur, who developed a vaccine against cholera, in 1897, and a vaccine against anthrax, in 1904 (Lombard et al., 2007). Nowadays, vaccination resorts to various strategies including subunit vaccines, based on the administration of antigenic agents capable of eliciting an immune response by the adaptive immune system that is protective against a specific pathogen.

The scientific advances, namely in genome sequencing, comparative proteomics and immunoinformatics algorithms, allowed the emergence of the concept of rationally designed peptide vaccines. Basically, this approach is based on rational sequence design of synthetic B-cell and T-cell epitopes that have a potential to trigger protective immune responses (Patronov and Doytchinova, 2013). The methodology of mapping the pathogen genome through computational tools is called reverse vaccination. Reverse vaccination presents four essential stages for the design process of a vaccine (Sette and Rappuoli, 2010; Kanampalliwar et al., 2013), namely:

1. Sequencing of the pathogen genome.

2. Computational analysis of the protein antigens that are expressed by the pathogen.

3. Prediction of epitopes.

4. Presentation of the candidate vaccine.

Epitope vaccines have some advantages over traditional vaccines, for instance the immune response affects conserved and immunogenic regions, allowing the elimination of pathogens with rapid evolution. Also, the insertion of multiple epitopes provides a more comprehensive response, and the specific stimulation of lymphocyte subpopulations, allows to develop selective immune responses. Finally, this new vaccine design approach is considered a safe and economical technology, since its peptide components, on the one hand, no adverse reactions are observed, for example fever, as verified in conventional vaccines based on pathogens and, on the other hand, have a simple composition and chemical stability that makes its production a relatively easy process (Li et al., 2014).

However, the design of rationally peptide vaccines implies two main challenges for the scientific community: firstly, individuals are genetically heterogeneous, so they have a different set of alleles, which can cause a different reaction to some peptides from a given pathogen, and, secondly, alleles are expressed at different frequencies in human populations. Therefore, the main objectives are maximizing coverage in the population to optimize the design of peptide vaccines and develop personalized therapies (Oyarzun and Kobe, 2015).

2.5 TUBERCULOSIS

The development of vaccines against pathogens is a challenging process, with *Tuberculosis (TB)* being one of the most studied cases since it is presently the infectious disease with higher mortality worldwide. TB is an infectious disease affecting all countries in the world, with a higher prevalence in less developed countries, particularly in Africa and Southeast Asia, as shown in **Figure 5** (Kaufmann, 2006). The WHO estimates that 1.8 million people die annually, making it one of the leading causes of death in the world (WHO, 2016).

Mycobacterium tuberculosis, the pathogen responsible for causing TB, was discovered by Koch (1982). This bacillus exhibits a complex infection cycle and numerous mechanisms to survive immune responses of the host, namely the ability to adopt a latent infection state until the individual's immune system becomes immunocompromised (Ernst, 2012; O'Garra et al., 2013).

The administration of live *Mycobacterium bovis bacillus Calmette-Gurin (BCG)* (Calmette et al., 1927), first used in 1921, is still the only vaccine available to prevent TB. However, this vaccine is only effective in newborns as a form of prevention of the most severe forms of TB in childhood and it has poor efficacy to prevent pulmonary tuberculosis in adults. The increasing rate of drug resistance due to the increase in strains of this bacterium, as well as the concomitant *Human Immunodeficiency Virus 1* (HIV-1) epidemic that accelerates the development of active TB, makes it urgent to develop an effective vaccine capable of reducing the incidence and mortality of the disease (Kaufmann, 2010). One of the major hypothesis under study to explain the lack of efficacy of BCG vaccine is the genetic diversity observed in *M. tuberculosis* namely at the level of some of its T cell epitopes (Coscolla et al.,

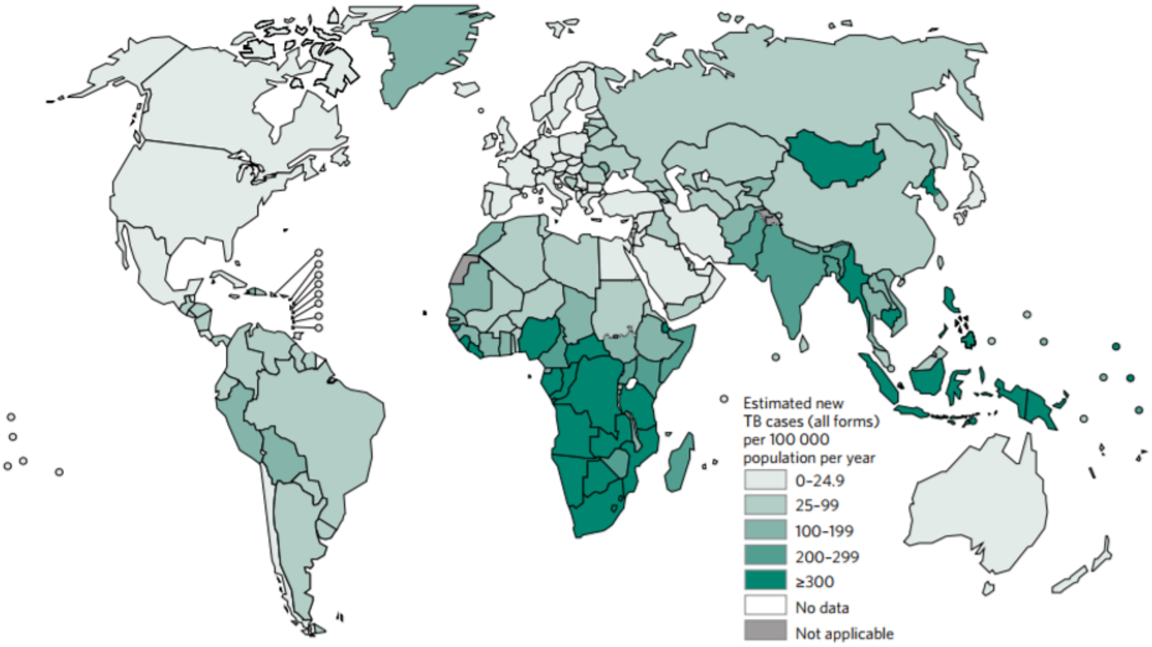

Estimated new TB cases (all forms) per 100 000 population per year
0–24.9
25–99
100–199
200–299
≥300
No data
Not applicable

Figure 5: Estimated tuberculosis incidence rates in 2015. Data source: Global Tuberculosis Report 2016. WHO], 2016.

2015; Osório et al., 2013). Thus, it is of large relevance in TB vaccine research to investigate the impact in host immune responses of varying *M. tuberculosis* epitopes.

2.6 EPITOPE PREDICTION TOOLS

The first step to rationally develop a peptide vaccine is the use of immunoinformatics resources to determine epitopes potentially capable of triggering an immune response. Currently, epitope prediction tools are available online and have contributed to optimize this in *in silico* stage. The most widely used and freely available tools in the field of immunoinformatics include:

- ***Immune Epitope Database and Analysis Resource (IEDB)***

  IEDB, in addition to the database, hosts a set of tools for prediction and analysis of immune epitopes. This method generates a quantitative prediction of the binding affinity between a peptide and an MHC molecule, based on *Position-specific scoring matrix (PSSM)* models and ML models. Although this database includes a variety of organisms, it only returns results concerning some alleles of HLA-A, HLA-B, HLA-DP and HLA-DQ. To mention also that the results obtained are related to a single sequence, not allowing to evaluate and to compare the impact of mutations (Kim et al., 2012).

- ***TEPITOPEpan***

  TEPITOPEpan is a method based on eleven libraries of PSSMs, corresponding to each of the HLA-DRB alleles obtained by biological experiments. This method performs only predictions of epitopes that bind to class II alleles, in particular for HLA-DRB (Zhang et al., 2012a).

- ***PREDIVAC***

  PREDIVAC is a framework developed to predict the binding of peptides to HLA class II, containing 95% of the allelic variants belonging to the HLA-DR locus. In addition to the prediction of $CD4^+$ T-cell epitopes in proteins, this tool makes a separate analysis of the population coverage (Oyarzún et al., 2013).

- ***Epitopemap***

  More recently, (Farrell and Gordon, 2015) developed a web application for integrated whole proteome epitope prediction, designated *Epitopemap*. This application returns results of multiple epitope prediction methods for any number of proteins in a genome, through visualization of plots, genome-wide analysis and estimates of epitope conservation. However, it focuses on the comparison of methods and on the differences between strains of the interest organism, not including any study on the immunogenic diversity of human populations, namely coverage calculation.

- ***NetMHCpan* and *NetMHCIIpan***

  NetMHCpan 3.0 (Nielsen and Andreatta, 2016) and *NetMHCIIpan* 3.1 (Andreatta et al., 2015) are two methods that generate quantitative predictions of peptide binding affinity to class I and class II HLAs, respectively. Some studies have shown that these tools were considered the best overall predictors in independent benchmarks (Chaves et al., 2012; Yang et al., 2005; Trolle et al., 2015). However, these tools return predictions on multi-column files that require processing and extraction of the relevant data.

The advance in that epitope prediction methods are clear and the presented tools are representative of the successful attempts to obtain a more efficient and flexible methods. However, these tools have some limitations. In particular, none of the tools allows an integrated comparison of predictions between wild type and mutant sequences, considering all overlapping peptides including the amino acid positions of interest. In addition, none of these methods includes information on statistical tests and population coverage calculation in an integrated way, which would be of great value for the rational design of T cell-based vaccination strategies for cancer, TB and other diseases (Oyarzun and Kobe, 2015).

2.7 NetMHCpan 3.0 and NetMHCIIpan 3.1

In recent years, the authors of these tools have implemented several versions and updates that aim to increase the accuracy in predicting this highly selective step. Despite some differences between these methodologies, the tools are based on the same theoretical basis. In this section, we present an overview and the optimization steps of these methods.

Firstly, the prediction tools were trained on two large distinct sets of quantitative MHC-peptide binding affinity measurements of different species. Data from both methods were obtained from the IEDB (Vita et al., 2015).

Both methods were implemented as a feed-forward artificial neural network ensemble, with each ANNs containing an input layer with 180 neurons, a hidden layer and an output layer with a single neuron (Nielsen et al., 2007), as shown in (**Figure 6**). An ensemble is a set of ML models, attempting to predict the output for the same set of inputs, and whose results are combined (e.g. using a voting scheme) to produce the final output.

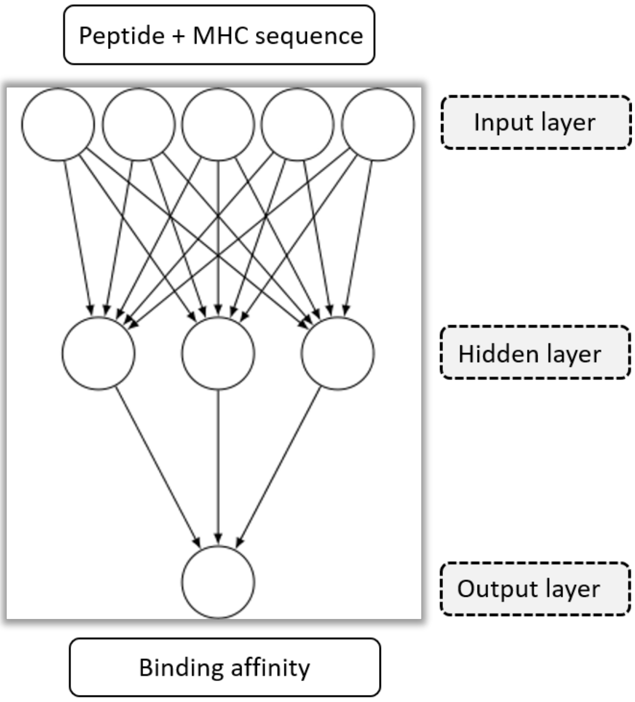


Figure 6: Feed-forward artificial neural network.

The input neurons consist of a set of peptides and MHC molecules which are presented as peptide sequences of 9 amino acids. These sequences were presented to the input layer of each network, using only BLOSUM encoding, where each amino acid is encoded using the vector of 20 scores for its match against all other amino acids, provided by the BLOSUM50 matrix (Henikoff and Henikoff, 1992). The BLOSUM50 is a member of the family of BLOSUM scoring matrices, each containing information about the similarity between each pair of amino acids, by assigning positive or negative values, respectively (**Figure 7**), being one of the most used in pairwise sequence alignments.

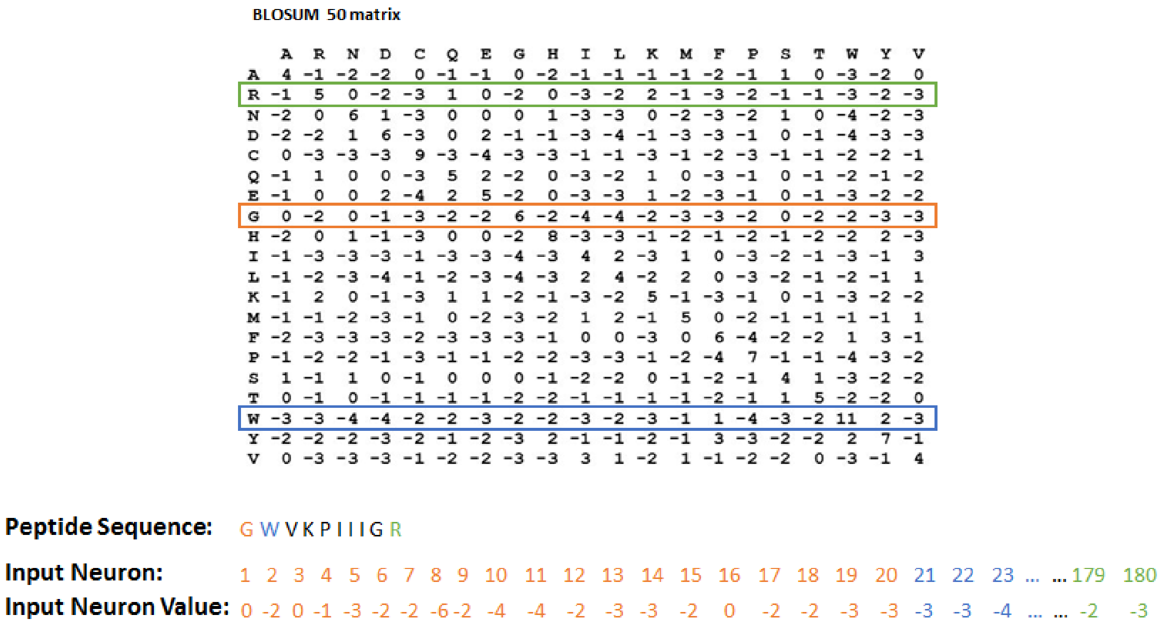
The measured binding affinities were rescaled between 0 and 1 to reach the output values of the neural networks. The transformation was performed through the following expression:

$$1 - \frac{log(aff)}{log(50000)}$$

where aff is the IC_{50} affinity value (nM) (Nielsen et al., 2003).

The set of ANNs was trained using a 5-fold cross validation, thus dividing the peptides into five training sets. All folds were balanced, being constituted by approximately the same number of strong binders, weak binders and non-binder peptides.

In each iteration, the training data was used to train the ANNs through the back-propagation algorithm, considering an initial random weight configuration, while the test data was used to define the stopping criterion for the network training (Baldi and Brunak, 2001). The back-propagation method was updated the network weights, in order to minimize the sum of quadratic errors between the predicted and measured binding affinity values (Nielsen et al., 2010).



BLOSUM 50 matrix

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Peptide Sequence: G W V K P I I I G R

Input Neuron: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 ... ... 179 180

Input Neuron Value: 0 -2 0 -1 -3 -2 -2 -6 -2 -4 -4 -2 -3 -3 -2 0 -2 -2 -3 -3 -3 -3 -4 ... ... -2 -3

Figure 7: The input peptide sequence was encoded by the BLOSUM50 scoring matrix, where each amino acid is represented by 20 neurons. Thus, the input layer consist of 180 neurons (for 9 aminoacids), which have the same color of the corresponding position in the input peptide.

This procedure was repeated several times using different initial values for the weights to construct the neural networks ensemble. *NetMHCpan* and *NetMHCIIpan* ensembles consist of 50 and 200 neural networks, respectively. Thus, the predicted binding affinity was calculated as the mean of the predictions obtained in the neural network ensembles.

In this way, the main objectives of the methods described above, for class I and class II HLAs, are to predict the strongest binding peptide of nine amino acids and to minimize the difference between predicted and measured binding affinity by updating the weights of the neural network.

3

WEB SERVER DEVELOPMENT

This chapter explores aspects related to the software development of our web server, covering both the design and implementation. We begin by considering an overview of its overall structure, emphasizing details of the architecture adopted, as well as the technology and main tools used in this project. After that, the technical aspects of implementation and organization in the web framework will be described. Finally, the process of integrating the epitope prediction tools in the back-end will also be addressed.

3.1 COMPUTATIONAL ARCHITECTURE

As previously reported, immunoinformatics is a growing area, which plays an increasingly important role in the *in silico* analysis of immunological data. In particular, epitope prediction tools generate huge amounts of data due to the high number of fragments originated following the digestion of a peptide sequence, as well as to the variety of existing HLAs. So, this task becomes complex for the researcher.

In this context, the present project describes the implementation of a web server that aims to simplify the study and selection of peptides capable of triggering an immune response. The web server, named HLA Binding Intelligence (HABIT), provides an easy-to-use and flexible interface, capable of integrating and comparing the prediction between wild type and mutant sequences. This will be achieved through interactive graphical visualization that will allow users to adapt the immunological data to the context of their projects.

3.1.1 *HABIT architecture and main tools*

This section describes the architecture of the web application, explaining how it is organized.

The whole implementation took as its base the *shiny* framework, which allows to create web-based applications, over the R scientific computing system.

The R language has been widely used in the process of data analysis and computational biology of many scientists. This project was developed in version 3.4.1 of R, since, besides being a free software, it presents strong graphical and statistical components. To implement the web application, we used the *shiny* v.1.0.5 package powered by RStudio.

Communication is one of the key elements in web development. In this case, it will be considered the communication between two distinct sides, the user side and the server side, as implemented in *shiny*. As the name implies, the user side refers to the web interface with which the user interacts, being responsible for requesting responses to the server and displaying them in the web browser.

On the other hand, the server side hosts the scripts that generate the web server, the *shiny* server software and the programs needed to generate the dynamic content to the user. All the back-end content was hosted in this project on a Linux (Ubuntu distribution) machine.

The HABIT, as shown in the **Figure** 8, is composed of four distinct modules:

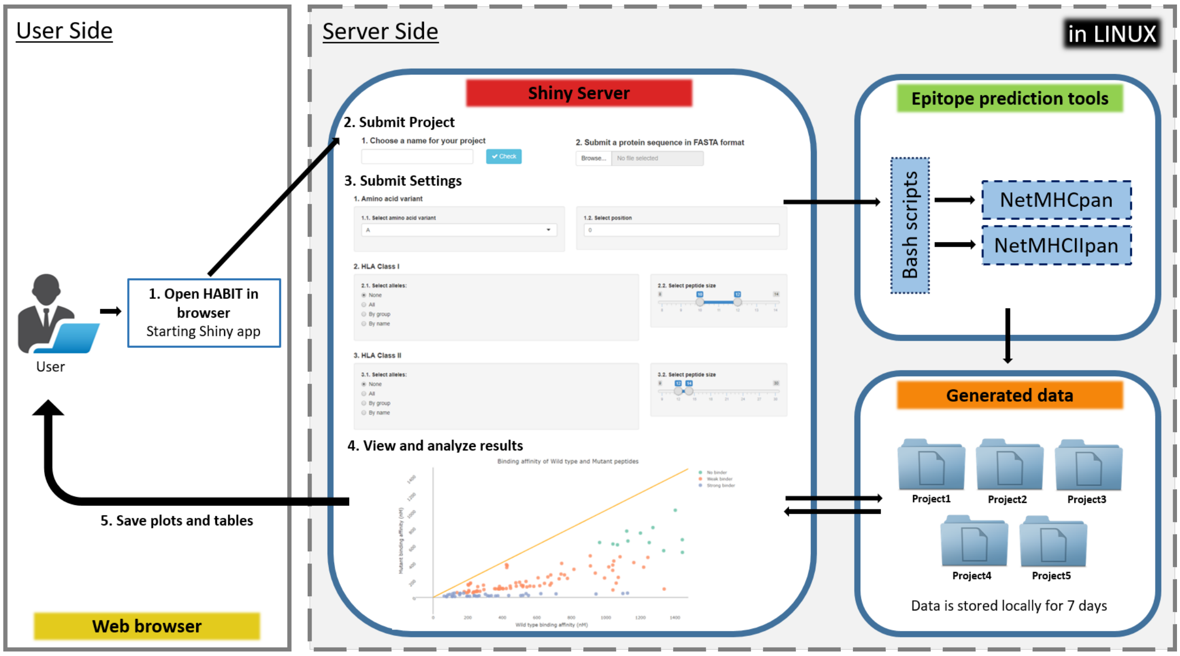


Figure 8: Web application architecture. HABIT is divided into the user side and the server side. The user side is composed only by the web browser module, while the server side presents three modules, which are the *shiny* server, the epitopes prediction tools, and the generated data.

A. Web Browser (user side)

This module is responsible for allowing user interaction with the HABIT framework through the web browser. The graphical interface ensures that the user will have access to all the features available on our web server.

As first step, the user chooses a name for her/his project, ensuring that the results will be saved in a unique and individual way, without needing login or authentication. To perform the study, the user submits the protein sequence in FASTA format, selects the amino acid variant and its position, as well as the set of class I and class II alleles. After the submission of the project, the data will be sent to the server, which will return the results.

B. **Shiny server**

HABIT is based on a set of scripts that provide the necessary guidelines for designing the application layout, validating user inputs, and monitoring outputs. The organization of the workflow scripts will be detailed later in this chapter.

To host our application on the web, we use the *shin*y open source software provided by RStudio. The *shiny* server is installed locally on our server, allowing the full management of our application.

This module plays a key role in the communication between the different modules of this web server. It is responsible for centralizing the handling of submissions from the web browser module, as well as sending messages due to errors or invalid information to the user.

The *shiny* server also assumes the function of requiring and optimizing the results obtained through epitope prediction tools. Finally, the generated data is saved and managed by this module.

C. **Epitope prediction tools**

As previously mentioned, *NetMHCpan* 3.0 and *NetMHCIIpa*n 3.1 tools were used in this project, in the back-end, to predict the peptide fragments with binding affinity to HLA class I and class II molecules, respectively.

NetMHCpan and *NetMHCIIpan* are command line tools available for the Linux operating system, so they require a local installation on our server and the development of input scripts in GNU bash on Linux.

D. **Generated data**

The generated data module is responsible for storing all the data coming from inputs and outputs of the processes triggered by HABIT. This module ensures that the data is stored separately for each user, guaranteeing their confidentiality. The user, through the name of the project that he chose initially, can upload and download the results. The data is stored locally on the server for a period of 7 days.

This module works in conjunction with the other web server modules. The web browser module provides the user interface, allowing the input of the project and visualization of the results. The *shiny* server module is responsible for uploading the data. Finally, the epitope prediction tools module generates the data that will be processed to produce the results.

3.2 WEB APPLICATION IMPLEMENTATION

3.2.1 *Shiny (from R studio)*

Shiny is a package available at Rstudio (Chang et al., 2017), which allows the development of interactive web applications. This framework is the basic structure for the implementation of this project, so a description of its organization and operation will be made.

This package is hosted on CRAN, and the current version is 1.0.5. In more detail, *shiny* is a simple and easy way for R users and data scientists to turn their data analysis into web applications. This is achieved since this framework is fully implemented in R, without requiring any knowledge in another language. However, the capabilities of *shiny* can be enhanced and complemented with CSS themes, HTML or JavaScript.

Shiny apps have two elements, a user-interface (UI) and a server. The UI component controls the layout and appearance of an application. In other words, it is responsible for telling *shiny*, through its own functions, how to organize the web page that will be shown to the user.

The server consists of the logical part of the application, as it contains the instructions that the computer needs to build the web page and to respond to the user's interactions with the page. Considering these two components, a possible way to define an application is to create two separate scripts files, "ui.R" and "server.R", respectively.

When the development phase of the application is completed, it is possible to make it available on the web through the *shiny* server software provided by Rstudio. The application can be hosted on the app developer's server or in the `shinyapps.io` plan.

For our project, it is important to highlight the self-hosted *shiny* server, since it was the option we used. This software, whose latest version is v1.5.4.869, is free and open source, requiring only the previous installation of the *shiny* server and a Linux server. Afterwards, the application will have a unique URL associated allowing users to access it in a web browser. More information about *shiny* can be found in `http://shiny.rstudio.com/`.

3.2.2 *Reactive programming*

HABIT was developed almost entirely in R, using a reactive structure characteristic of the *shiny* package. This reactive structure has demonstrated an increasing value in the development of interactive graphs implemented on the web, since it allows the application to dynamically update the input values when changed by the user. The understanding of this reactive programming model is fundamental to create a more efficient application, taking full advantage of the available resources.

Shiny applications are composed of specific functions that must be called in the user interface ("ui.R") script and developed in the server side script ("server.R") to achieve the outputs shown to the user. Each created function has an associated input value, which is indicated as `input$variableName`, and an output expression, which is represented by `print(input$variableName)`. In this way, each input value will have associated an output expression, creating a link through the unique identifier they share.

The reactivity system is implemented from three classes of objects, which are the reactive values, the reactive expressions, and the observers. Reactive values are the source of the system and correspond to the input values that are detected by the other reactive objects. The reactive expressions act as conductors between the sources and the endpoints of the reactive processes and are used to make the operation more efficient. The observers refer to the endpoints, that is to the output expressions of a function, which have the objective of accessing the reactive values and reactive expressions.

All these objects can not be considered normal variables in R, because these are characterized by being changed according to the input value of the user. Each reactive object has an associated unique symbol, the representation is present in the **Figure 9**.

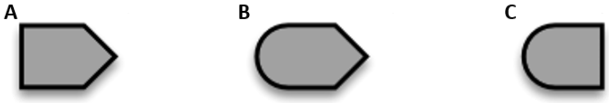


Figure 9: Objects in reactive programming. **A.** Reactive value. **B.** Reactive expression. **C.** Observer.

Thus, it is expected that when the user enters a new `input$variableName` there is an update of the associated output expression. However, this is not an instantaneous process. In fact, the update process only occurs when the `print(input$variableName)` is executed again, so *shiny* has an alert system to detect which observers need to be re-executed. Although this procedure is quite fast, it is an approach that involves reserving an R session when the application is started, since it is used to monitor and re-execute expressions.

Considering the reactive programming system used by *shiny* applications, debugging and tracking are sometimes complicated and challenging. To facilitate these processes, the `Reactive Log Visualizer` function is able to track the operation of the application, encompassing information about which objects are being executed, which dependencies are between reactive objects and which is the running time.

3.2.3 *R scripts workflow*

In this section, we will describe and explain how the workflow of R scripts that constitute HABIT is organized into custom functions and separate files. Usually, *shiny* applications, as described previously, are defined only by two scripts, "ui.R" and "server.R", which are responsible for the user interface and server components, respectively. However, due to the increasing complexity of our web server, to provide a more robust and intuitive approach, we have developed the script organization displayed in **Figure 10**.

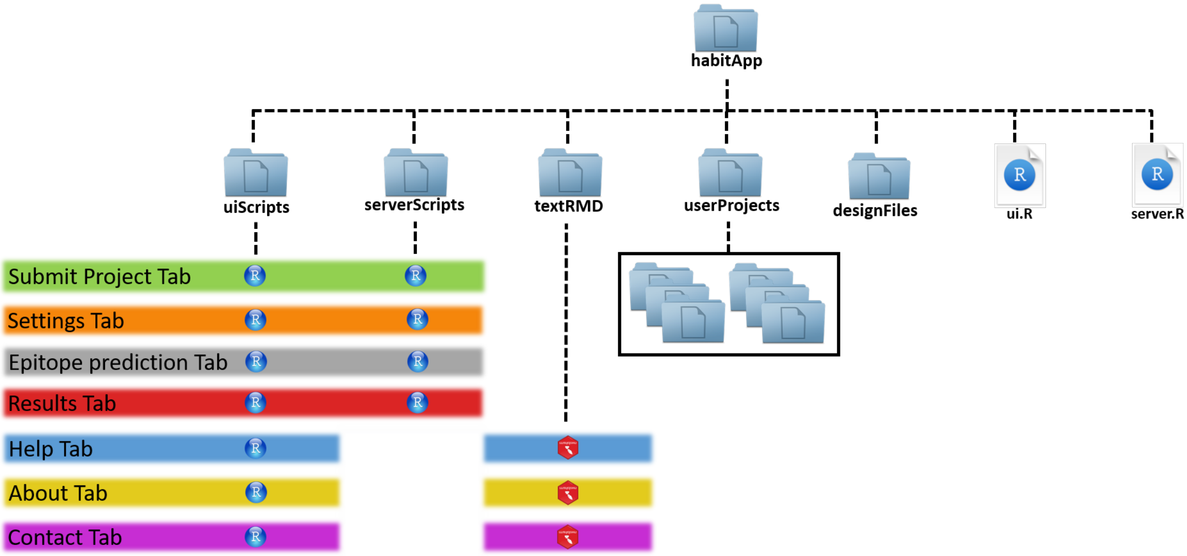


Figure 10: HABIT code organization. All files and folders necessary for proper web server operation are stored in the main directory, *habitApp*. The main scripts, "ui.R" and "server.R", are both stored in this directory so that R can recognize and launch the application. The "uiScripts", "serverScripts", and "textRMD" subfolders contain all files that are complementary to the main scripts. All projects created on the web server are stored in *userProjects* folder. Finally, the files relating to the design and appearance of HABIT are present in *designFiles*.

The main scripts, "ui.R" and "server.R", of an application were kept in our workflow, thus ensuring that RStudio automatically recognizes these files to generate a *shiny* application. Both scripts, as well as all files and folders that are required by the application, should be stored in a single directory, in this project called *habitApp*.

The HABIT header consists of seven distinct tabs, which are: *Submit Project*, *Settings*, *Epitope prediction*, *Results*, *Help*, *About* and *Contact*. The described tab separation was used as the basic criterion for dividing the main scripts, so that the implemented code had a more simplified structure and a more efficient management.

In general, this procedure consisted of removing the code corresponding to each tab, initially present in the main scripts, and placing in new individual scripts, always considering the division “ui.R” and “server.R”. The code associated with each tab is accessed by the main scripts.

For this, the tabs that relate to the submission and visualization steps of a project have two associated files, constituted by the functions UI and server, respectively. For example, as outlined in the **Figure 10**, the *Submit Project* tab has a “ui_SubmitProject.R” script and a “server_SubmitProject.R” script. The same line of thinking is applied to the remaining tabs of this group. Differently, the information tabs (*Help*, *About* and *Contact*) present only the UI script and are complemented with an R Markdown file.

To organize the workflow of scripts generated through this methodology, and taking into account that R only recognizes the main scripts, the remaining code files have been placed in 3 different directories (*uiScripts*, *serverScripts* and *textRMD*), that store all UI scripts, server scripts and R Markdown files, respectively.

The main features of each script are shown below:

- **Main UI**

  The main UI script is responsible for the general structure and interface that is presented in the first instance to the user. In other words, it is the script that includes the front-end code. The UI module is built from a web framework and the implementation of functions developed specifically for the *shiny* package. Taking into account the current web standards, this code is complemented with simple classes of HTML and CSS, in order to make dynamic and improve the graphical interface of our application.

  In this file, we also include the necessary instructions to place the HABIT logo and create the navigation bar with the tabs at the top level of the web page. This R script also includes the directory of the user interface script files for each tab, allowing RStudio to recognize and access them as complementary user interface.

- **Main Server**

  The main server script encompasses the instructions that RStudio needs to build the web browser and respond to user submissions. When the user interacts with the interface, the server receives a message to send the new data to the interface. In this case, the reactivity process is verified through a `show/hide` function that is displayed in the header to inform the user of the need to start a project before continuing the search for the tabs.

  This script also includes the path to the server files that contains information about each tab.

- **Submit Project tab**

  The HABIT homepage has the *Submit project* tab selected, so the two scripts that contain the elements and functions necessary for the user to proceed with the first submission step. These scripts are responsible for ensuring the uniqueness and authenticity of each project, since they create a folder with the name that the user chose for their work. Through a `show/hide` function, the user is notified with a message about the availability of the chosen name. In the created folder, the file will be saved in the FASTA format with the wild type sequence that the user submits.

In addition to the user being able to submit a new project, in this tab, he can access a project that has been created previously and, thus, access the results. For this purpose, he/she only need to enter the name of that project. Also, the user can load an example project that is available to give an overview of the HABIT application.

- **Settings tab**

  The second tab concerns a crucial step in the study of possible epitopes belonging to a given protein sequence, named as *Settings* tab. This tab is divided into two sections, the *Basic* settings, containing the mandatory fields, and the *Advanced* settings, which are composed of the default parameters of the epitope prediction tools.

  The UI script provides a set of *shiny* widgets, which are web elements that users interact with, such as *selectInput*, *numericInput*, *checkboxGroupInput*, and *sliderInput*. These collect values entered by the user to send them to the server. In the *Basic* section, the user, based on the widgets, selects which amino acid variant and position, as well as which class I and class II alleles of interest and the size range of the peptides. The *Advanced* section allows the user a more personalized search, which consists of four *numericInputs* that correspond to two *NetMHCpan* and two *NetMHCIIpan* threshold values. These parameters will be explained in detail in the next section.

  After submitting the input data entered by the user into the widgets, the server script collects the values and creates two bash scripts that run *NetMHCpan* and *NetMHCIIpan*. To perform this, R proceeds in a series of steps to process the data to be read by the epitope prediction tools, namely:

  1. Reads the protein sequence in FASTA format that was submitted previously (see example represented in the **Figure 11**);
  2. Locates the position of the amino acid variant in the sequence (example: position 30) and for each peptide size entered by the user (example: size 10), creates a peptide sequence that contains all possible peptides including the mutant amino acid (example: between positions 21 and 39);
  3. For each peptide sequence, creates the corresponding mutant sequence, substituting the amino acid variant at the indicated position (example: R → L). Each pair of sequences is stored in a FASTA file;
  4. Saves the selected alleles in a text file. Each HLA class has an individual file.

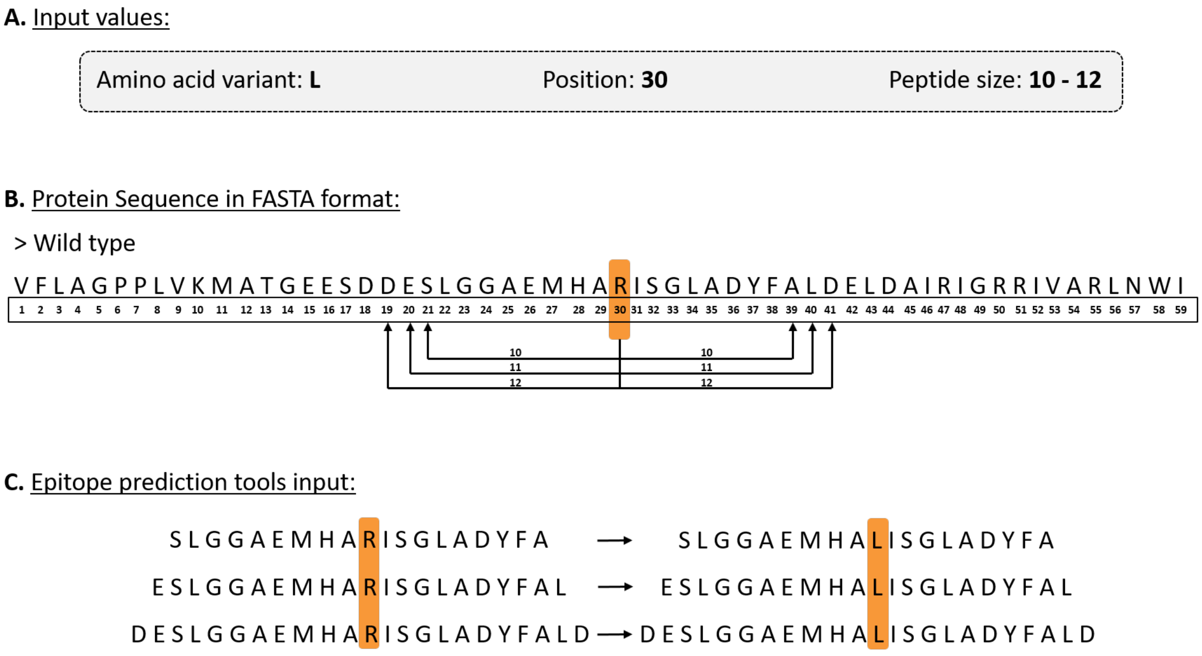

A. Input values:
Amino acid variant: L
Position: 30
Peptide size: 10 - 12
B. Protein Sequence in FASTA format:
> Wild type
VFLAGPPLVKMATGEESDDESLGGAEMHARISGLADYFALDELDAIRIGRRIVARLNWI
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
10
11
12
10
11
12
C. Epitope prediction tools input:
SLGGAEMHARISGLADYFA → SLGGAEMHALISGLADYFA
ESLGGAEMHARISGLADYFAL → ESLGGAEMHALISGLADYFAL
DESLGGAEMHARISGLADYFALD → DESLGGAEMHALISGLADYFALD

Figure 11: Processing and filtering the input data. **A.** In the *Settings* tab, the user fills a set of parameters necessary for the study, namely selects the amino acid variant, position, and range of peptide size. **B.** From the wild type sequence submitted in FASTA format, the application locates the input position and will extract the sequences which, for each chosen peptide size, contain all peptides that include the mutant amino acid. **C.** In the last step, the algorithm creates the mutant sequences corresponding to each of the wild type sequences.

- **Epitope prediction tab**

  The *Epitope prediction* tab works as the link between the end of a submission and the release of the results, while the server performs the management and organization process to obtain the outputs from the epitope prediction tools. This tab was developed to inform the user that her/his project was submitted correctly, and to return an estimate of the time that the analysis process will take.

  The output of these tools does not have a fixed running time, which can be influenced by different issues, namely the HLA class being predicted, the size of the chosen peptide, the local server and the availability of the tool. After analysing the time frame obtained by each class for a single peptide size, we decided to consider an average time of three seconds and four seconds for *NetMHCpan* and *NetMHCIIpan*, respectively. Thus, each of these tools, within the stated time, predicts the binding affinity of a HLA molecule for all peptides of the same size that it is possible to digest in a peptide sequence.

  Based on the pre-defined prediction times, the server script estimates the time it takes to perform the entire process of predicting epitopes, and consequently, for the user to access the results. It is important to highlight that the tools work on individual parallel systems, so if a submission includes analysis of the two classes, the time for the most time-consuming analysis is returned.

- **Results tab**

  The scripts associated with the *Results* tab are the most important and complete in this HABIT implementation process, since they filter and process the raw data of the prediction tools, as well as their subsequent application in the construction of graphs and tables, being easier to interpret the results.

  The UI script is responsible for presenting the results obtained to the user, ensuring that they are always separated according to the class of HLA molecules. In each class, the results are distributed by three new tabs:

  - Data Visualization: Presents two scatter plots that make a global analysis of the data.

  - Data Analysis: Consists of a summary table with the most relevant results of each allele analysis, including comparisons between the wild type and mutant sequences and statistical analysis.

  - Population Coverage: Contains the frequencies of each HLA molecule based on different geographic areas. These values were obtained from the IEDB database.

  Simultaneously, the back-end system performs the exploratory analysis of the data, from the filtering and treatment to the exposure of the results in the browser. First, the outputs, after preprocessing to remove purely informational lines about the prediction, are read by R as a single data frame. In this phase, we use some specific R functions to handle data frames, namely `aggregate`, `which`, `subset`, `rbind`, among others. These functions have associated parameters that allow us to specify columns and rows to access, select or delete variables, and consequently, return a new data structure with the unique values of those parameters. With these resources, we were able to extract the most relevant results of the study quickly and efficiently.

  The graphics were entirely developed using the *plotly* package available by R. This is a graphical library capable of creating high-quality interactive and custom graphics. Thus, the interactive behavior of the generated graphs allows the user a total control, namely to approach/remove a certain area, select visualization filters or even add/remove data. Thus, automatic updating of plots is a powerful, sophisticated and flexible tool for the exploration and study of data in the field of immunoinformatics.

  The R package, designated *DataTables (DT)*, was used to display the data treated in tables. *DT* provides several features, such as sorting and searching.

- **Help tab**, **About tab** and **Contact tab**

  R Markdown is an R package for writing dynamic documents that combine simple text and R code. R Markdown (.Rmd) files can be converted to different formats, such as PDF, web page (HTML), among others.

  These last three tabs have an informative character, so they only need a UI script to expose fixed content to the user. In order to facilitate and streamline the text formatting process, the text present in each of the tabs was written in a R Markdown and later converted to Markdown (.md).

The multiple splitting of scripts across each tab allowed us to work with simpler and more specific scripts, since they are made up of fewer lines of code. Each script has a specific directory which makes searching and resolving possible errors easier. Thus, this approach and scripting division allowed to increase the productivity and effectiveness in the development of the application.

3.3 EPITOPE PREDICTION TOOLS INTEGRATION

MHC molecules exhibit high specificity, recognize and bind only to a small portion of peptides submitted to the antigen presentation pathway, in other words, antigens that trigger an immune response. This property is fundamental to understand the process of mediating the immune response, therefore, several studies have been performed to develop methods able to predict the peptides that bind to these molecules.

Considering the two classes of MHC, class I and class II, and after careful analysis, *NetMHCpan* 3.0 and *NetMHCIIpan* 3.1 were the epitopes prediction tools chosen to integrate into our web server pipeline. These prediction methods are property of the Center for Biological Sequence Analysis, Technical University of Denmark. Both are freely available for academic users through their open source licenses available at `http://cbs.dtu.dk/`.

3.3.1 *NetMHCpan and NetMHCIIpan features*

One of the main challenges of MHC genes concerns to the huge amount of polymorphism existing within and between species. In the case of our web server, only the allelic variants belonging to the HLA loci will be considered, which encompasses the alleles found in humans.

In the context of HABIT, *NetMHCpan* makes predictions for all peptides with a length of 8-14 amino acid for the wild type and mutant sequences that include the mutant amino acid. The prediction of possible epitopes can be applied to a set of 2924 alleles, consisting of 6 distinct HLA groups, which are HLA-A, HLA-B, HLA-C, HLA-E and HLA-G.

On the other hand, *NetMHCIIpan* is in charge of predicting peptide binding affinity to HLA class II, it is able to test peptides with a length between 9 and 30 mers. The server allows to predict 660 alleles belonging to the HLA-DR loci.

3.3.2 *Management system: from bash scripts to outputs*

This section describes how the epitope prediction tools were integrated into our web server through the Linux operating system.

Both tools are installed locally on our Ubuntu server as standalone software packages. *NetMHCpan* and *NetMHCIIpan* work only through the command line, in this way a parallel system to the *shiny* server was developed, always considering the different characteristics and parameters between the two tools. This system is entirely executed by the shell in a series of steps (**Figure 12**), ensuring that the R session is occupied as little time as possible.

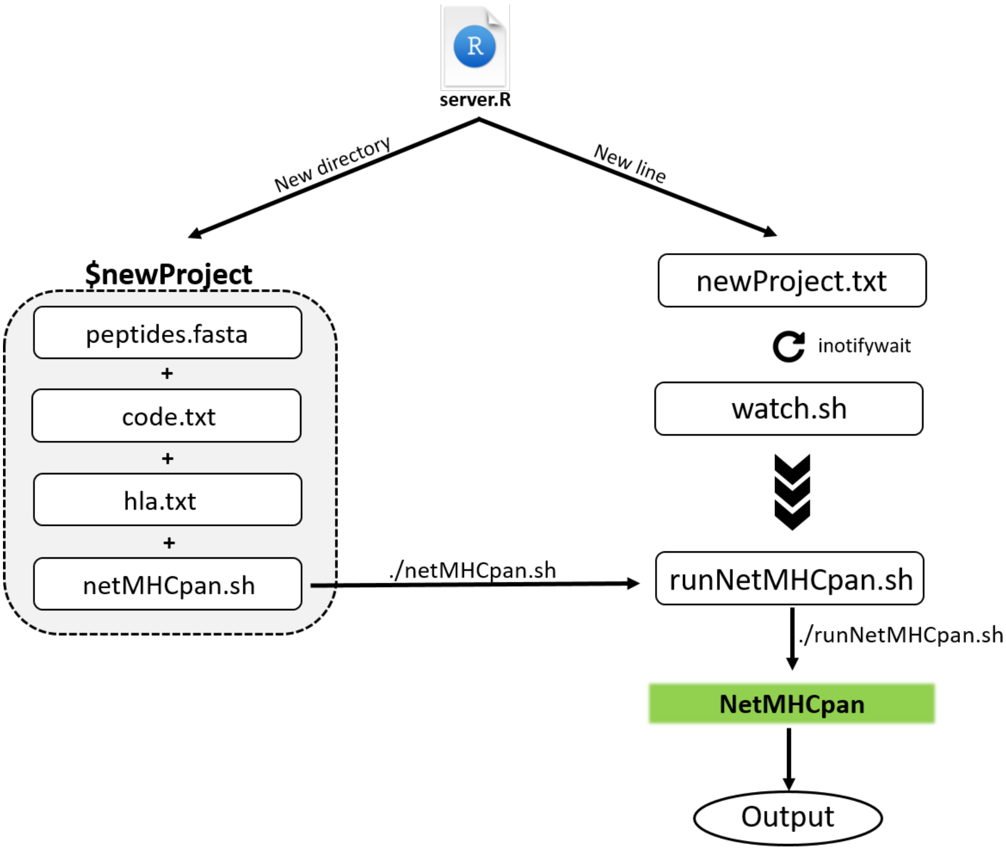


Figure 12: *NetMHCpan/NetMHCIIpan* management system. In a first instance, R organizes the user input files and connects to the implemented system. This system is autonomous and fully executed in the Linux shell, which aims to launch the tools of prediction of epitopes to generate the outputs.

The initialization of this system is achieved from two fundamental steps performed by the server script corresponding to the *Settings* tab. Firstly, it is responsible for placing the following files in the project folder under analysis:

- **peptides.fasta** - FASTA files that contain each sequence pair (wild type and mutant) depending on the size. Explained above in **Figure 11**;
- **code.txt** - This file is created by R and corresponds to the model of the commands that run *NetMHCpan*, that is, contains only the fixed parameters;
- **hla.txt** - includes the HLA list that the user selected;
- **netMHCpan.sh** - This is an equal template script for all projects, whose purpose is to create the final script that runs netMHCpan. From the "code.txt" are added the fields relating to a peptide and a HLA. Considering this, the tools are launched from the "runNetMHCpan.sh" script, which contains specific arguments to customize the output for each project submitted. The lines of code that constitute it are based on:

```
# NetMHCpan:
$ ../netMHCpan -a hla.txt -f file.fasta -l 10 -rth 0.5 -rlt 2 > output.txt

# NetMHCIIpan:
$ ../netMHCIIpan -a hla.txt -f peptides.fasta -length 10 -affS 50.0 -affW
   500.0 -rankS 0.5 -rankW 2 > output.txt
```

The selected options concern:

-a: HLA molecule name.

-f: Input file in FASTA format.

-l / -length: Peptide length. Peptide sequences will be digested into peptided with the specified length.

-affS: Binding affinity in IC_{50} of strong binders peptides.

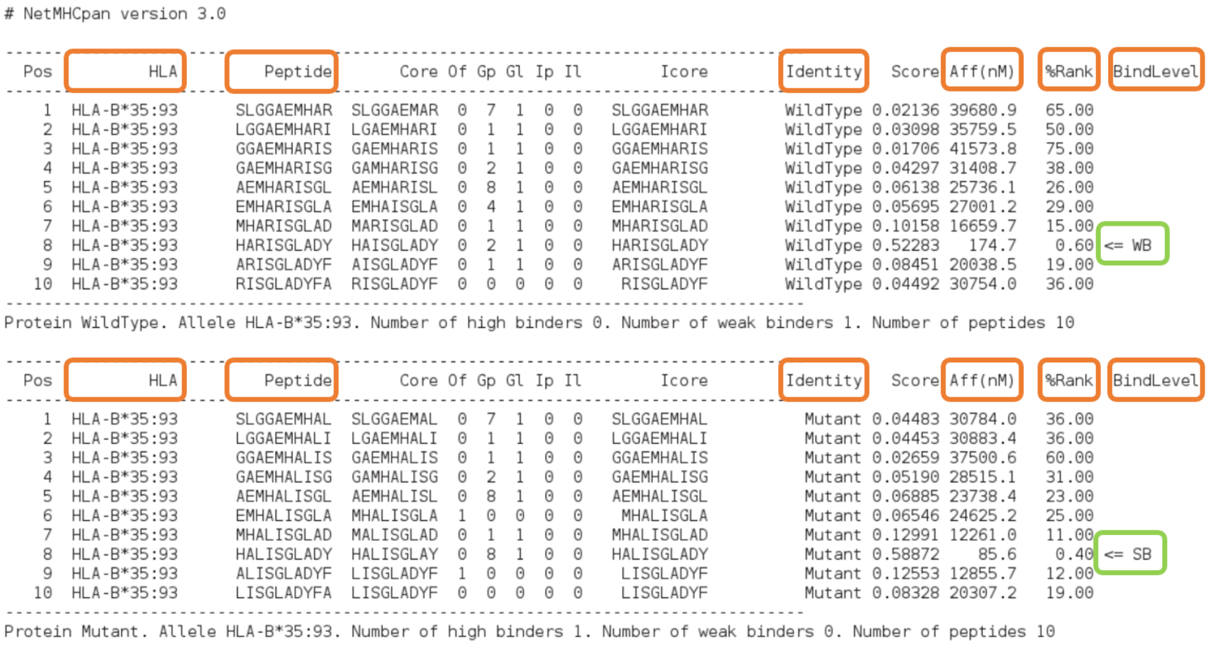
-affW: Binding affinity in IC_{50} of weak binders peptides.

-rth / -rankS: Rank threshold for strong binders peptides.

-rlt / -rankW: Rank threshold for weak binders peptides.

Secondly, and after the shell runs the “netMHCpan.sh” script, R adds a new line with the project name in the “newProject.txt” file. After all these operations are performed by R, the system is ready to initialize. The implemented system is administered through a shell script, designated “watch.sh”, which monitors the “newProject.txt” file. For this purpose, `inotifywait` was used, which is a Linux kernel feature capable of monitoring specific files and alerting the system when a change, such as deletion or insertion, is verified. Thus, whenever the system detects a change in the file, it runs the “runNetMHCpan.sh” script, accessing *NetMHCpan* and returning the outputs.

NetMHCpan and *netMHCIIpan* have different outputs, as can be seen from the **Figure 13** and the **Figure 14**, respectively. For our web server, each output contains the information and data relating to the unique HLA and a single size for a particular wild type/mutant pair of a peptide sequence. These outputs are constituted by a series of columns, and only a few will be used to generate results, among which we highlight:

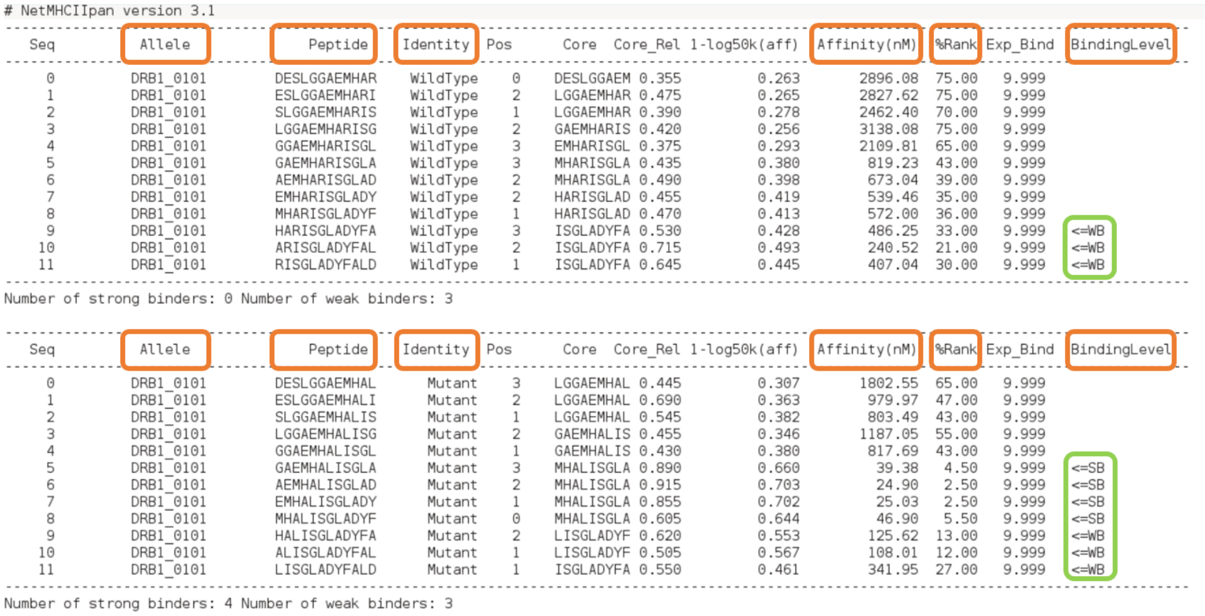


```
# NetMHCpan version 3.0

---------------------------------------------------------------------------------------------------
  Pos          HLA         Peptide       Core Of Gp Gl Ip Il         Icore  Identity  Score Aff(nM) %Rank BindLevel
---------------------------------------------------------------------------------------------------
    1  HLA-B*35:93      SLGGAEMHAR  SLGGAEMAR  0  7  1  0  0    SLGGAEMHAR  WildType 0.02136 39680.9  65.00
    2  HLA-B*35:93      LGGAEMHARI  LGAEMHARI  0  1  1  0  0    LGGAEMHARI  WildType 0.03098 35759.5  50.00
    3  HLA-B*35:93      GGAEMHARIS  GAEMHARIS  0  1  1  0  0    GGAEMHARIS  WildType 0.01706 41573.8  75.00
    4  HLA-B*35:93      GAEMHARISG  GAMHARISG  0  2  1  0  0    GAEMHARISG  WildType 0.04297 31408.7  38.00
    5  HLA-B*35:93      AEMHARISGL  AEMHARISL  0  8  1  0  0    AEMHARISGL  WildType 0.06138 25736.1  26.00
    6  HLA-B*35:93      EMHARISGLA  EMHAISGLA  0  4  1  0  0    EMHARISGLA  WildType 0.05695 27001.2  29.00
    7  HLA-B*35:93      MHARISGLAD  MARISGLAD  0  1  1  0  0    MHARISGLAD  WildType 0.10158 16659.7  15.00
    8  HLA-B*35:93      HARISGLADY  HAISGLADY  0  2  1  0  0    HARISGLADY  WildType 0.52283   174.7   0.60 <= WB
    9  HLA-B*35:93      ARISGLADYF  AISGLADYF  0  1  1  0  0    ARISGLADYF  WildType 0.08451 20038.5  19.00
   10  HLA-B*35:93      RISGLADYFA  RISGLADYF  0  0  0  0  0     RISGLADYF  WildType 0.04492 30754.0  36.00
---------------------------------------------------------------------------------------------------
Protein WildType. Allele HLA-B*35:93. Number of high binders 0. Number of weak binders 1. Number of peptides 10

---------------------------------------------------------------------------------------------------
  Pos          HLA         Peptide       Core Of Gp Gl Ip Il         Icore  Identity  Score Aff(nM) %Rank BindLevel
---------------------------------------------------------------------------------------------------
    1  HLA-B*35:93      SLGGAEMHAL  SLGGAEMAL  0  7  1  0  0    SLGGAEMHAL    Mutant 0.04483 30784.0  36.00
    2  HLA-B*35:93      LGGAEMHALI  LGAEMHALI  0  1  1  0  0    LGGAEMHALI    Mutant 0.04453 30883.4  36.00
    3  HLA-B*35:93      GGAEMHALIS  GAEMHALIS  0  1  1  0  0    GGAEMHALIS    Mutant 0.02659 37500.6  60.00
    4  HLA-B*35:93      GAEMHALISG  GAMHALISG  0  2  1  0  0    GAEMHALISG    Mutant 0.05190 28515.1  31.00
    5  HLA-B*35:93      AEMHALISGL  AEMHALISL  0  8  1  0  0    AEMHALISGL    Mutant 0.06885 23738.4  23.00
    6  HLA-B*35:93      EMHALISGLA  MHALISGLA  1  0  0  0  0     MHALISGLA    Mutant 0.06546 24625.2  25.00
    7  HLA-B*35:93      MHALISGLAD  MALISGLAD  0  1  1  0  0    MHALISGLAD    Mutant 0.12991 12261.0  11.00
    8  HLA-B*35:93      HALISGLADY  HALISGLAY  0  8  1  0  0    HALISGLADY    Mutant 0.58872    85.6   0.40 <= SB
    9  HLA-B*35:93      ALISGLADYF  LISGLADYF  1  0  0  0  0     LISGLADYF    Mutant 0.12553 12855.7  12.00
   10  HLA-B*35:93      LISGLADYFA  LISGLADYF  0  0  0  0  0     LISGLADYF    Mutant 0.08328 20307.2  19.00
---------------------------------------------------------------------------------------------------
Protein Mutant. Allele HLA-B*35:93. Number of high binders 1. Number of weak binders 0. Number of peptides 10
```

Figure 13: *NetMHCpan* 3.0 output. The example concerns all 10-mer peptides which include the amino acid of interest for the wild type and mutant sequence. The binding affinity was predicted for the HLA-B35:93 allele. The columns with the orange box contain the data used by the HABIT. The cells with the green box identify the epitopes classified as a strong binder or a weak binder.

- **HLA / Allele** - Allele name;
- **Peptide** - Amino acid sequence of the potential epitope;
- **Identity** - Annotation of the input sequence, wild type or mutant;

- **Aff(nM) / Affinity(nM)** - Predicted binding affinity in nanoMolar IC_{50};
- **% Rank** - Express percentage rank of predicted affinity compared to a set of 200.000 random natural peptides;
- **BindLevel** - Indicates whether a peptide is a *Strong binder (SB)* or a *Weak binder (WB)*.



```
# NetMHCIIpan version 3.1
-----------------------------------------------------------------------------------------------------------------------------
  Seq          Allele            Peptide   Identity Pos       Core  Core_Rel 1-log50k(aff) Affinity(nM) %Rank Exp_Bind BindingLevel
-----------------------------------------------------------------------------------------------------------------------------
    0       DRB1_0101       DESLGGAEMHAR   WildType   0   DESLGGAEM 0.355          0.263       2896.08  75.00   9.999
    1       DRB1_0101       ESLGGAEMHARI   WildType   2   LGGAEMHAR 0.475          0.265       2827.62  75.00   9.999
    2       DRB1_0101       SLGGAEMHARIS   WildType   1   LGGAEMHAR 0.390          0.278       2462.40  70.00   9.999
    3       DRB1_0101       LGGAEMHARISG   WildType   2   GAEMHARIS 0.420          0.256       3138.08  75.00   9.999
    4       DRB1_0101       GGAEMHARISGL   WildType   3   EMHARISGL 0.375          0.293       2109.81  65.00   9.999
    5       DRB1_0101       GAEMHARISGLA   WildType   3   MHARISGLA 0.435          0.380        819.23  43.00   9.999
    6       DRB1_0101       AEMHARISGLAD   WildType   2   MHARISGLA 0.490          0.398        673.04  39.00   9.999
    7       DRB1_0101       EMHARISGLADY   WildType   2   HARISGLAD 0.455          0.419        539.46  35.00   9.999
    8       DRB1_0101       MHARISGLADYF   WildType   1   HARISGLAD 0.470          0.413        572.00  36.00   9.999
    9       DRB1_0101       HARISGLADYFA   WildType   3   ISGLADYFA 0.530          0.428        486.25  33.00   9.999  <=WB
   10       DRB1_0101       ARISGLADYFAL   WildType   2   ISGLADYFA 0.715          0.493        240.52  21.00   9.999  <=WB
   11       DRB1_0101       RISGLADYFALD   WildType   1   ISGLADYFA 0.645          0.445        407.04  30.00   9.999  <=WB
-----------------------------------------------------------------------------------------------------------------------------
Number of strong binders: 0 Number of weak binders: 3

-----------------------------------------------------------------------------------------------------------------------------
  Seq          Allele            Peptide   Identity Pos       Core  Core_Rel 1-log50k(aff) Affinity(nM) %Rank Exp_Bind BindingLevel
-----------------------------------------------------------------------------------------------------------------------------
    0       DRB1_0101       DESLGGAEMHAL     Mutant   3   LGGAEMHAL 0.445          0.307       1802.55  65.00   9.999
    1       DRB1_0101       ESLGGAEMHALI     Mutant   2   LGGAEMHAL 0.690          0.363        979.97  47.00   9.999
    2       DRB1_0101       SLGGAEMHALIS     Mutant   1   LGGAEMHAL 0.545          0.382        803.49  43.00   9.999
    3       DRB1_0101       LGGAEMHALISG     Mutant   2   GAEMHALIS 0.455          0.346       1187.05  55.00   9.999
    4       DRB1_0101       GGAEMHALISGL     Mutant   1   GAEMHALIS 0.430          0.380        817.69  43.00   9.999
    5       DRB1_0101       GAEMHALISGLA     Mutant   3   MHALISGLA 0.890          0.660         39.38   4.50   9.999  <=SB
    6       DRB1_0101       AEMHALISGLAD     Mutant   2   MHALISGLA 0.915          0.703         24.90   2.50   9.999  <=SB
    7       DRB1_0101       EMHALISGLADY     Mutant   1   MHALISGLA 0.855          0.702         25.03   2.50   9.999  <=SB
    8       DRB1_0101       MHALISGLADYF     Mutant   0   MHALISGLA 0.605          0.644         46.90   5.50   9.999  <=SB
    9       DRB1_0101       HALISGLADYFA     Mutant   2   LISGLADYF 0.620          0.553        125.62  13.00   9.999  <=WB
   10       DRB1_0101       ALISGLADYFAL     Mutant   1   LISGLADYF 0.505          0.567        108.01  12.00   9.999  <=WB
   11       DRB1_0101       LISGLADYFALD     Mutant   1   ISGLADYFA 0.550          0.461        341.95  27.00   9.999  <=WB
-----------------------------------------------------------------------------------------------------------------------------
Number of strong binders: 4 Number of weak binders: 3
```

Figure 14: *NetMHCIIpan* 3.1 output. The example concerns all 12-mer peptides which include the amino acid of interest for the wild type and mutant sequence. The binding affinity was predicted for the HLA-DRB1*01:01 allele. The columns with the orange box contain the data used by the HABIT. The cells with the green box identify the epitopes classified as a strong binder or a weak binder.

Each peptide submitted to the *NetMHCpan* is ranked based on the associated rank percentile. While, in the *NetMHCIIpan* tool, the epitopes are classified according to one of two parameters, affinity value or rank percentile. Initially, a threshold value is defined for each of the parameters, allowing to classify the peptides as strong binder, weak binder or non-binder. However, these parameters are independent, i.e. only one of them needs to be within the defined criteria.

Regardless of a peptide to be classified based on the affinity value or percentile rank, the lower the number associated with each of these parameters, the greater the binding affinity of the peptide to the HLA. By default, the following values are considered as threshold values by the epitope prediction tools:

$$\textbf{High affinity peptides:} \quad SB < 50nM \quad \text{or} \quad SB < 0.5\%$$

$$\textbf{Intermediate affinity peptides:} \quad 50nM < WB < 500nM \quad \text{or} \quad 0.5\% < WB < 2\%$$

$$\textbf{Low affinity peptides:} \quad 500nM < Non\ binder < 5000nM \quad \text{or} \quad Non\ binder > 2\%$$

3.4 HABIT AVAILABILITY

The present work proposes a web platform able to facilitate and accelerate the discovery of molecular determinants that influence the variation in immune responses mediated by T cells, thus contributing to the study of immunotherapy.

HABIT is available in a free online version for academic users at `http://192.168.104.46:3838/HABIT/` (before public release access to the tool is only available inside the University of Minho's network). Our web server is a useful tool for researchers who do not have advanced programming skills for data analysis, but also for users who want to get a fully automated analysis in a personalized and integrated way through a simple and user-friendly interface.

The final HABIT pipeline is shown in the **Figure 15**. This includes the organization and visualization structure of our web server, as well as the set of steps present in each of the three main tabs, which are *Submit Project*, *Settings* and *Results*.

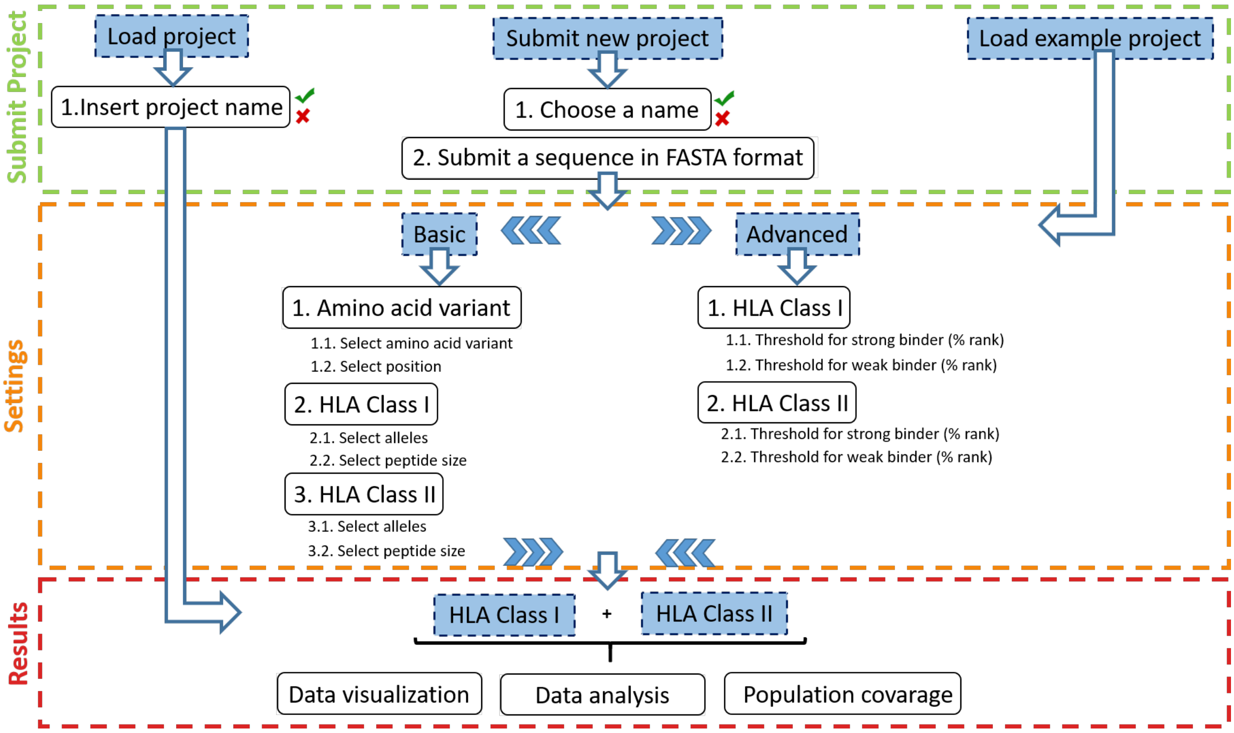


Figure 15: Schematic representation of the HABIT pipeline.

4

RESULTS AND DISCUSSION

HABIT was developed with the purpose of generating and interpreting, in an integrated way, *in silico* results of peptide-HLA binding predictions. The computational architecture and methods described in the previous chapter were used to implement the features and resources that will be demonstrated and explained in detail in this chapter. In addition, to test and validate our web server, we have chosen a specific case study that will also be analysed and discussed.

4.1 INTEGRATION AND VISUALIZATION OF PEPTIDE-HLA BINDING PREDICTIONS

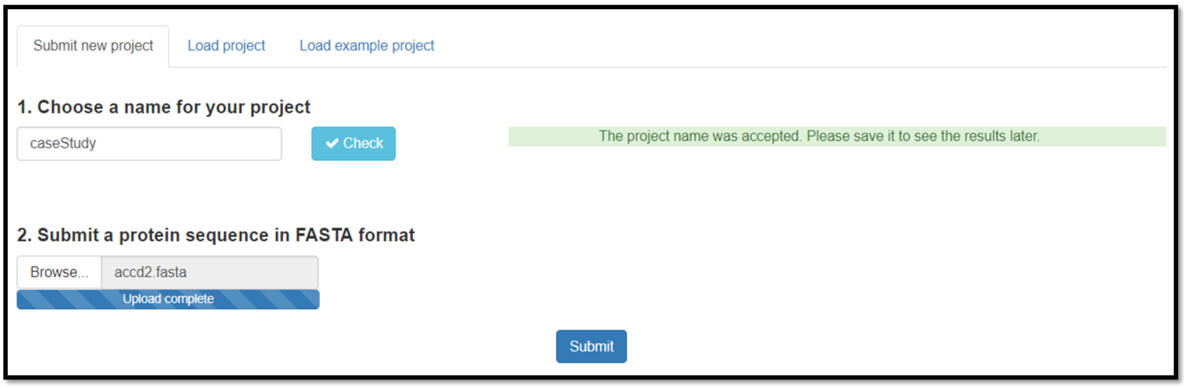
T cell-mediated immune responses are triggered by the recognition of specific peptides that bind to HLA proteins. The amino acid variants in the epitopes may affect the HLA-peptide interaction and consequently the level and type of generated T cell responses. Thus, being able to predict HLA-binding epitopes constitutes a key step for immunotherapy. Whereby, a variety of immunoinformatic approaches have been developed, however, there is still no tool capable of analysing the impact of amino acid variants on T-cell antigenicity. Since these tools do not perform comparative tasks, the researcher needs computational programming skills to filter and analyse the large amount of data generated by the epitope prediction algorithms.

Considering the above-described problem, the main purpose of HABIT implementation was to provide a comparison of the results of the individual prediction of wild type and mutant sequences based on the selection of all overlapping peptides including amino acid positions of interest. In addition, subsequent data prioritization strategies, such as statistical tests or the calculation of population coverage, were also included.

4.1.1 *Submit new project*

The first step to take advantage of the resources of our web server is to create a new project, which requires the selection of specific features and the filling of some mandatory fields. To create a project, the user chooses a name to be associated, and then the system will check its availability, with one of the following messages being returned: "The project name was accepted. Please save it to see the results later." or "Please choose another name, it already exists.". Since HABIT does not request login or authentication for the user to access, the name of the project is essential to manage each user session and to associate a project with a user. Thus, this name has two distinct purposes, on the one hand, the epitope prediction tools integrated in our web server may take some time to return the outputs, so the user after the submission can close the browser and access again when the results are complete. On the other hand, the results are available for consultation during a period of 7 days. In both cases, the access to the project is done through the *Load project* tab.

The next step is to load a file of a protein sequence in FASTA format. It is only necessary that the input file contains the wild type sequence, since the mutant sequence is created automatically by the server. This procedure is shown in the **Figure 16**.



1. Choose a name for your project

caseStudy

The project name was accepted. Please save it to see the results later.

2. Submit a protein sequence in FASTA format

accd2.fasta

Upload complete

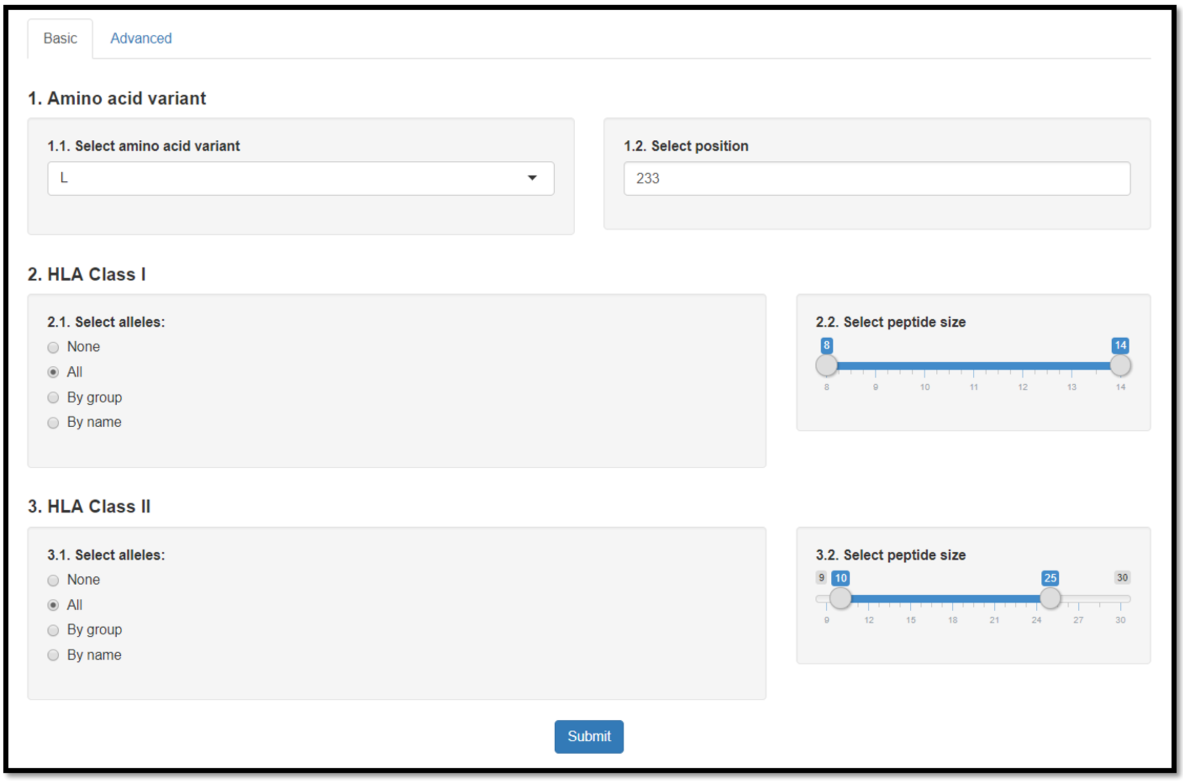
Figure 16: Start panel corresponding to the *Submit Project* tab. This step allows the user to choose the name associated with the project and upload a FASTA file with the protein sequence.

4.1.2 *Manage settings: basic and advanced*

After the submission and validation of the new project, it is necessary to specify the parameters that customize and restrict the analysis process that aims to compare the wild type with the mutant sequences and, consequently, to realize the impact that a substitution of a single amino acid can cause in the interaction of the predicted epitope with an HLA molecule.

The *Settings* tab was divided into two sections, basic and advanced, as shown in **Figure 17**. The first, containing basic definitions, presents mandatory parameters, including the information about the amino acid variant or the selection of class I and class II alleles fields. At this point, the user must select:

- **Amino acid variant:** Corresponds to the amino acid of interest responsible for inducing the mutation in a sequence. This is the only amino acid that differs between the wild type and mutant sequences.
- **Position:** Refers to the position where the amino acid variant is in the sequence.
- **Alleles:** Associated with both classes of alleles, including several options for selection of these proteins ("None", "All", "By group", "By name").
- **Peptide size:** Range of peptide lengths in which the query protein will be fragmented.



1. Amino acid variant

1.1. Select amino acid variant

L

1.2. Select position

233

2. HLA Class I

2.1. Select alleles:

○ None
● All
○ By group
○ By name

2.2. Select peptide size

8 – 14

3. HLA Class II

3.1. Select alleles:

○ None
● All
○ By group
○ By name

3.2. Select peptide size

10 – 25

Figure 17: Basic interface of *Settings* tab, depicting selection for AccD2 protein. In one single screen, the user can take a decision on what amino acid variant and both HLA classes to study.

Regarding the advanced settings tab, it includes optional fields corresponding to the thresholds values for strong binder and weak binder based on the % rank value. These parameters are used later by the epitope prediction tools.

The interface implemented for the submission stage is extremely intuitive and user-friendly, allowing the user to incorporate their study into HABIT quickly through a few clicks in the web browser. This approach aims to simplify the whole process so that the researcher must indicate only the information strictly necessary to obtain results previously treated.

4.1.3 *Epitope prediction: running time and obtaining outputs*

The implementation of this step aims to link the submission of a project and the access to the results obtained from the raw data provided by the epitope prediction tools. The epitope prediction tab has the main function to inform the user that their project was correctly submitted and the input data were accepted. In the meantime, the server is responsible for managing the data and executing the integrated tools on the web server, as well as filtering the outputs and performing the implicit calculations to return the results to the user.

Considering that outputs of the epitope prediction tools may take some time to conclude, depending on the number of HLA molecules and the size of the peptides that the user selected, we chose to return only the estimated time. This implementation allowed increasing the capacity and effectiveness of HABIT, since the R session started in each application is occupied as little time as possible.

As expected, it is important to take into account that larger numbers of HLA molecules and selected peptide sizes increase the time to obtain the results. Currently, the maximum execution time possible for a submission on our web server is about 15 hours. Once the process of each session has elapsed, the results are available to be consulted by the researcher.

4.1.4 *Interactive graphical interface*

The most important component, which encompasses the main objective of the development of this thesis, concerns the interactive graphical interface. The purpose of an researcher to perform peptide/HLA-binding predictions can range from a global immunotherapy study to a highly specific research. The common point is that the researchers need the results in an effective graphical representation that summarises the results obtained from a wide range of data. Therefore, the following sections were designed: *Data visualization*, *Data analysis* and *Population coverage*.

Firstly, the *Data visualization* tab consists of two fully interactive scatter plots, which allow the freedom and flexibility to customize the contents of these graphs through a checkbox that includes:

- **Axis:** In an epitope prediction study, there are two essential quantitative parameters to be analysed, the % rank and the binding affinity. This option allows the user to change the plot axis between % rank and binding affinity. By default, % rank is the selected option.;

- **Alleles of interest:** After processing all the data, HABIT automatically selects the 20 alleles in which a greater impact of the mutation was verified, that is, the alleles that had a greater increase of the binding affinity when comparing wild type to mutant peptides. Thus, the graphs are initially constructed with data from these 20 alleles but the user can at any time add or remove HLA alleles according to its preference.

The graphs developed to be part of the presentation of results in our web server are a quick and intuitive way of visualizing the global distribution of the data and make a first general comparison of the verified differences between the wild type and mutant peptides.

The first graph allows to interpret the tendency of the predicted epitopes with a ML approach. Each observation corresponds to a pair of wild type/mutant peptides which were predicted for a specific HLA, i.e. between each pair of peptides only a single amino acid change was verified. Thus, each dot has an X coordinate corresponding to the % rank value of the wild type peptide and a Y coordinate corresponding to the % rank value of the mutant peptide. In the graph, there is also an orange line that symbolizes the bisector of odd quadrants, whereby all observations overlapping the line show equal % rank values in the wild type and mutant peptides. In this way, the distance to the line allows to realize the general impact of the mutation, in which the points present above the line indicate an increase in the peptide-HLA binding capacity, whereas the points below the line represent that the mutation was responsible for increasing the affinity between them. When overlapping the mouse at a point the following information is indicated: the allele, the peptide sequence, and the % rank value for wild type and mutant, respectively. The hover text that appears provides an instant indication of the epitopes in any instance of the graph.

Also in this tab, HABIT organizes the results into a second plot, focusing only on the strong binding peptides. This graph is primarily intended to evaluate the differences between the wild type and mutant peptides, where at least one is considered a strong binder. Thus, for each wild type or mutant peptides belonging to this cluster, the corresponding wild type or mutant peptide is also added to the plot, regardless of its classification with respect to the binding affinity. In the graph, the X-axis corresponds to the % rank values, the Y-axis corresponds to the selected alleles and the orange line the threshold value for the strong binding peptides, by default 0.5%. Each observation has a set of associated information that can be seen by overlapping the mouse, namely the allele, the peptide sequence and the % rank value.

These graphs, developed through the package *plotly* allow interaction with graphical observations using the zoom in and zoom out features, making the information more understandable and focused on the regions of interest. The user can download the generated graphics at any time as an image file. Considering the amount of data generated from an epitope prediction tool, HABIT's interactive graphical visualizations represent a technology that adds huge value to the web server, since it allows users through visual analysis to explore in greater detail observations and identify possible patterns.

The *Data analysis* tab complements the interactive plots and consists of a table that includes the analysis of data obtained from the output files of the epitope prediction tools after a filtering. The data are grouped to facilitate their interpretation. The first column, labelled “Allele”, contains the list of HLA molecules previously selected by the user. The “Wild type” and “Mutant” columns make a general and individual analysis of the wild type and mutant peptides, respectively. These two columns are subdivided into:

- **Predicted SB Epitopes** - number of peptides considered to bind with high affinity to the HLA molecule, according to the thresholds specified in *Advanced* settings. The default setting considers strong binding peptides when the % rank value is less than 0.5%;
- **Mean binding affinity (nM)** - corresponds to the sum of all predicted binding affinity values for a specific HLA molecule and division by the total number of peptides overlapping variant amino acid position;
- **Mean % rank** - corresponds to the sum of all predicted % rank values for a specific HLA molecule and division by the total number of peptides overlapping variant amino acid position.

The following column, “Wild type vs Mutant”, as its name suggests, presents the results obtained by comparing the behavior of the two peptide sequences within each allele, analysing the following parameters:

- **% Predicted epitopes (mutant - wild type)** - returns the difference between the number of mutant and wild type peptides considered strong binders regarding the percentage of predicted peptides;

$$\text{\% Predicted epitopes} = \frac{\text{mutant SB peptides - wild type SB peptides}}{\text{peptides overlapping variant amino acid position}} \times 100$$

- **Mean binding affinity (nM) (t-test p-value)** - corresponds to the p-value calculated from the t-test to compare the difference between the mean binding affinity values of the wild type and mutant peptides. This was performed to evaluate whether the differences between these two groups are statistically significant;
- **Mean % rank (t-test p-value)** - corresponds to the p-value calculated from the t-test to compare the difference between the mean % rank values of the wild type and mutant peptides. This was performed to evaluate whether the differences between these two groups are statistically significant.

The last column in this table, designated the “World population coverage”, includes the frequencies of each HLA gene corresponding to the world population. In order to complete the information provided in this column, the third section of the interactive results interface, “Population coverage”, presents a table with the frequencies of each allele in 15 geographic regions.

In general, the last two sections described provide the user with a set of data encompassing all peptides tested by each HLA molecule, allowing for an integrated analysis including individual and comparative details on the wild type and mutant sequences as well as values underlying statistics and frequency in different regions of the globe. Initially, alleles are presented in alphabetical order but can be sorted by any other column. Both tables can be exported in Excel or PDF format.

4.2 CASE STUDY

Several factors, including the lack of an effective vaccine and the emergence of drug-resistant *Mycobacterium tuberculosis*, threaten the control of TB. Every year, this infectious disease is responsible for the death of millions of people around the world. Thus, the development of effective immunotherapy strategies for TB is essential and urgent. One of the stronger hypothesis to explain the lack of efficacy of the only vaccine currently in use for TB is the genetic variability in the population of human infecting *M. tuberculosis*.

Understanding the impact of *M. tuberculosis* antigenic variants in triggering host T-cell mediated immune responses may lead to the design of new and more efficient vaccines against TB. The group were this thesis was developed is addressing this problematic and has previously used immunoinformatic tools to characterize the impact of variants in known *M. tuberculosis* epitopes (Yruela et al., 2016). Ongoing *in vitro* and *in vivo* studies from the group (Magalhaes et al, unpublished) have strongly suggested the existence of potential novel M. tuberculosis T cell epitopes in a specific genotype of *M. tuberculosis*. This epitope is harboured in the AccD2 gene.

AccD2 encodes a 529 amino acid protein as represented in the gray box below. This gene is still poorly described in the literature and few features are known. It is likely that AccD2 encodes an enzyme involved in the degradation of fatty acids (Gande et al., 2004; Ehebauer et al., 2015).

To demonstrate the usefulness of HABIT, we have studied the predicted impact of the R233L that is present in all Lineage 1 *M. tuberculosis* bacteria, and was found to be under diversifying selection, thus is likely to confer an evolutive advantage to the bacteria in particular contexts (Magalhães et al. unpublished). This mutation results from the substitution of a residue of arginine (R) by a leucine (L) at position 233.

In this work, HABIT was used to address the impact of this amino acid substitution of the predicted HLA-binding properties of the AccD2 peptides. Considering the R233L mutation, we proceeded to fill the settings that best fit this study. For the amino acid variant, we selected the letter L as the amino acid of substitution and also selected position 233 which refers to the location of the arginine in the wild type sequence.

```
>M. tuberculosis H37Rv|Rv0974c|AccD2
VLQSTLDPNASAYDEAAATMSGKLDEINAELAKALAGGGPKYVDRHHARGNLTPRERIELLVDPDSPFLELSPLA
AYGSNFQIGASLVTGIGAVCGVECMIVANDPTVKGGTSNPWTLRKILRANQIAFENRLPVISLVESGGADLPTQK
EIFIPGGQMFRDLTRLSAAGIPTIALVFGNSTAGGAYVPGMSDHVVMIKERSKVFLAGPPLVKMATGEESDDESL
GGAEMHARISGLADYFALDELDAIRIGRRIVARLNWIKQGPAPAPVTEPLFDAEELIGIVPPDLRIPFDPREVIA
RIVDGSEFDEFKPLYGSSLVTGWARLHGYPLGILANARGVLFSEESQKATQFIQLANRADTPLLFLHNTTGYMVG
KDYEEGGMIKHGSMMINAVSNSTVPHISLLIGASYGAGHYGMCGRAYDPRFLFAWPSAKSAVMGGAQLSGVLSIV
ARAAAEARGQQVDEAADAAMRAAVEGQIEAESLPLVLSGMLYDDGVIDPRDTRTVLGMCLSAIANGPIKGTSNFG
VFRM
```

We took advantage of HABIT capacity to effectively handle large amounts of data, to perform the most comprehensive and complete analysis possible. In the HABIT molecules sections, we selected the "All" option for both classes, as a result the analysis was performed using 2924 and 660 class I and class II HLA molecules, respectively. Subsequently, we selected all peptides ranging in size from 8 to 14 mers for the class I molecules and all peptides consisting of 12 to 20 amino acids for the class II molecules.

Regarding the *Advanced* settings tab that includes threshold values for the definition of strong and weak binding peptides, they were submitted with the default values of rank $<$ 0.5% for strong binders and $<$ 2% for weak binders.

Since we decided to perform a large-scale study, selecting all the available HLA alleles and peptides sizes, the results of the analysis of the AccD2 protein took about 14 hours and 37 minutes to complete.

4.2.1 *Impact of the amino acid variant on binding affinity to class I alleles*

NetMHCpan made the prediction for all peptides overlapping the position 233 that was a total of 450296 peptides being analysed. For each HLA molecule, binding affinity was predicted for 145 peptides, of which 77 were *wild type* peptides and the remainder were the respective mutant peptides.

In the first instance, HABIT evaluates the overview of the results through a summary which counts the number of peptides considered weak and strong binders. The results show differences between the *wild type* and mutant sequences in what regards to class I HLA binding. 1633 *wild type* peptides were classified as strong binding and 4647 peptides as weak binding, while in the mutant sequence the predictions found 2031 strong and 5784 weak binders. As a first impression, obtained solely from this summary, the introduction of this mutation into the AccD2 encoded peptide seems to somehow increase the affinity to one or more class I HLA molecules. Furthermore, HABIT provided more in-depth results that allowed quickly obtaining the answer to two fundamental questions: “What are the HLA molecules that have increased binding affinity?” and “Are these differences between wild type and mutant sequences statistically significant?”.

The **Figure 18 A** shows the first graph for the AccD2 protein with all the observations of 20 HLA molecules which show a higher binding affinity for mutant peptides. The representation clearly highlights that the majority of dots is located below the orange line, indicating lower values of % rank for the mutant peptides meaning a higher predicted ability to bind to the highlighted alleles. The largest cluster of peptides is classified as non-binder, i.e., peptides which have no predicted binding affinity for the HLA molecules. The zoom in tool was used to highlight only the area that includes the points classified as weak and strong binding peptides (**Figure 18 B**). In this perspective, we can visualize a trend for a decrease in the value of the % rank predicted for the mutant peptides but it is also evident that this occurs only in a small percentage of the peptides tested.

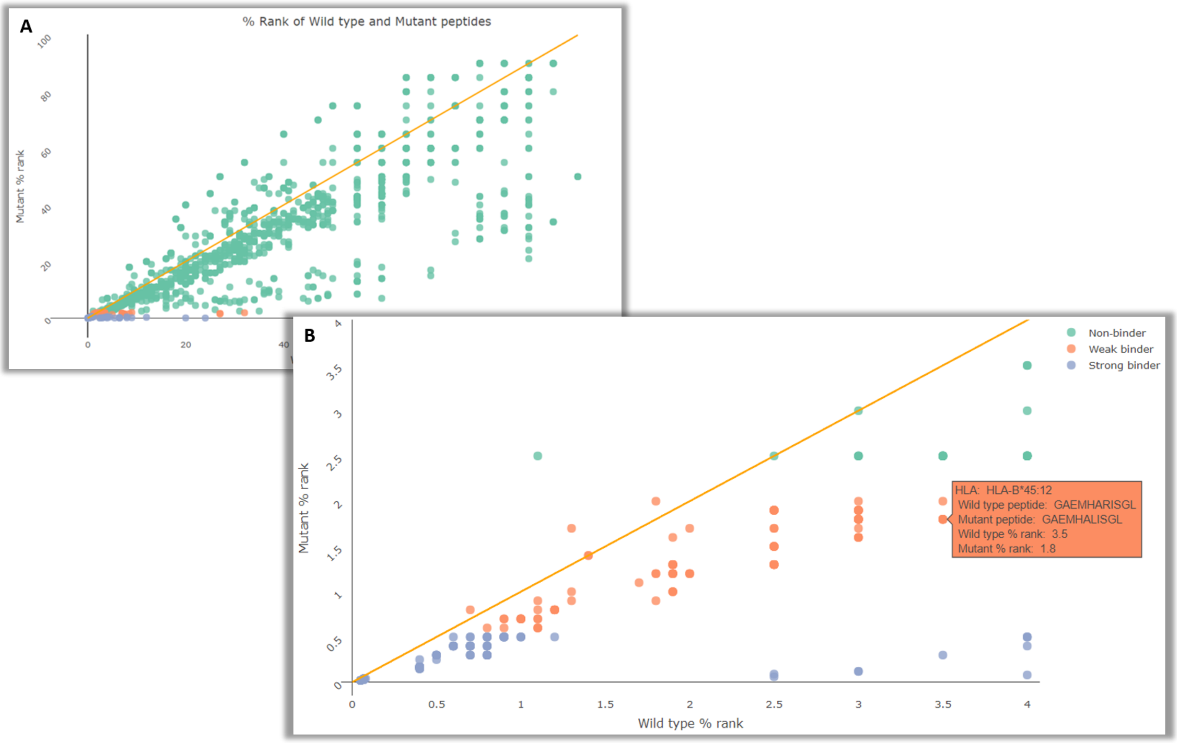


Figure 18: Interactive plot that compares the % rank values of each peptide pair (wild type/mutant) belonging to the AccD2 protein when tested for an HLA molecule. Considering that the greater the distance from an observation to the orange line the larger the difference between the pairs of peptides. **A.** The plot includes all observations of the 20 alleles which showed higher binding affinity to the mutant peptides. **B.** The graph obtained by using the zoom in tool includes only the points of greatest interest to the study of the accd2 protein.

The second dot plot generated by HABIT was also very useful to further interpret these results (**Figure 19**). The distribution of the points corroborates the conclusion drawn from the previous graph, that is, only a minority of the more than 400 thousand tested peptides was classified as. It should be noted that for each HLA molecule there are at most 5 mutant peptides predicted to binding with high affinity, representing a very reduced fraction of the tested peptides.

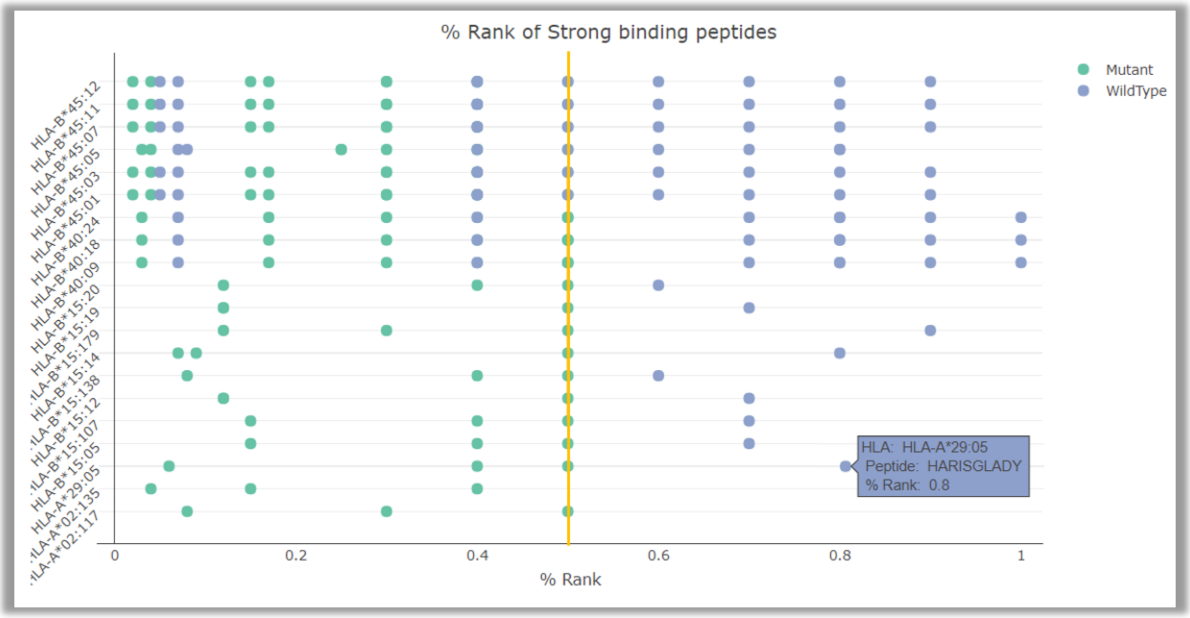
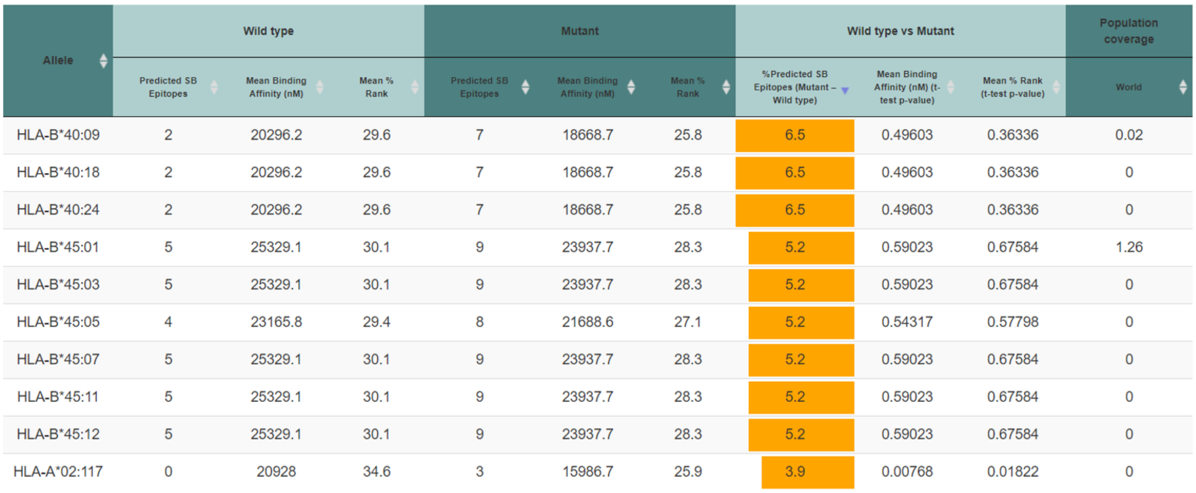


Figure 19: Interactive dot plot that selects all pairs of peptides (wild type/mutant) belonging to the AccD2 that have at least one strong binding peptide. By allele, this plot allows comparison of the % rank values of two peptides and identification of similarity between them. The orange vertical line overcomes the limit of stong binding peptides.

All the data from the context-specific epitopes were placed in the **Table 1**, being organized according to the allele for which the binding affinity prediction was performed. The table describes the information concerning the top 10 alleles that demonstrate a higher rate of variation in binding affinity values between the *wild type* and mutant epitopes. However, based on the column "% Predicted epitopes (mutant - wild type)" we can verify that the mutation only induced a maximum binding affinity increase of 6.5% for 3 HLA molecules (HLA-B*40:09, HLA-B*40:18 and HLA-B*40:24). Furthermore, the p-values of both t-tests are greater than 0.05, which indicates that the differences found between the wild type and mutant peptides were not statistically significant at the 95% confidence level. Moreover, the percentage of world population coverage that is associated with these three alleles is very low.

Overall, the automated analysis performed by HABIT highlighted that this region is not likely to bind any of the 2924 class I HLA molecules tested and that the R233L mutation

Table 1: Individual and comparative analysis of the outputs obtained from the epitope prediction tools for the wild type and mutant sequences. The results were calculated for each selected class I allele and sorted, in the table, in decreasing order of column "% Predicted epitopes (mutant - wild type)".



Allele	Wild type			Mutant			Wild type vs Mutant			Population coverage
	Predicted SB Epitopes	Mean Binding Affinity (nM)	Mean % Rank	Predicted SB Epitopes	Mean Binding Affinity (nM)	Mean % Rank	%Predicted SB Epitopes (Mutant – Wild type)	Mean Binding Affinity (nM) (t-test p-value)	Mean % Rank (t-test p-value)	World
HLA-B*40:09	2	20296.2	29.6	7	18668.7	25.8	6.5	0.49603	0.36336	0.02
HLA-B*40:18	2	20296.2	29.6	7	18668.7	25.8	6.5	0.49603	0.36336	0
HLA-B*40:24	2	20296.2	29.6	7	18668.7	25.8	6.5	0.49603	0.36336	0
HLA-B*45:01	5	25329.1	30.1	9	23937.7	28.3	5.2	0.59023	0.67584	1.26
HLA-B*45:03	5	25329.1	30.1	9	23937.7	28.3	5.2	0.59023	0.67584	0
HLA-B*45:05	4	23165.8	29.4	8	21688.6	27.1	5.2	0.54317	0.57798	0
HLA-B*45:07	5	25329.1	30.1	9	23937.7	28.3	5.2	0.59023	0.67584	0
HLA-B*45:11	5	25329.1	30.1	9	23937.7	28.3	5.2	0.59023	0.67584	0
HLA-B*45:12	5	25329.1	30.1	9	23937.7	28.3	5.2	0.59023	0.67584	0
HLA-A*02:117	0	20928	34.6	3	15986.7	25.9	3.9	0.00768	0.01822	0

in the AccD2 protein was not predicted to induce large alterations in the HLA-binding properties since no statistically significant differences were found.

4.2.2 *Impact of the amino acid variant on binding affinity to class II alleles*

The peptide binding predictions of AccD2 peptides to class II HLA molecules was also entirely performed using HABIT. This analysis encompassed all candidates for epitopes which contained the overlapping region of amino acid variant R233L resulting in the analysis of a total of 190080 peptides. In detail, binding affinity was predicted for 288 peptides (144 *wild type* and 144 mutant) for each of the 660 selected class II HLA alleles.

As a first analysis, and based on the initial summary described in HABIT, 12224 *wild type* peptides and 22887 mutant peptides were classified as weak binders, that is, the R233L mutation increased the number of weak binding peptides to almost double. With an even greater impact, the number of peptides classified as strong binders increased by about 12 times, for the *wild type* sequence only 270 peptides were predicted as strong binders, while for the mutant sequence a total of 3221 peptides were considered as having strong HLA-binding capacity.

In the *Data visualization* tab, the two scatter plots were shown to the 20 class II HLA alleles, automatically selected by HABIT as being the ones with higher differences between mutant and *wild type* sequences (**Table 2**).

The plot comparing the binding affinity values per each wild type/mutant peptide pair for a specific HLA (**Figure 20 A**), presents the general distribution for all the observations of

Table 2: Top 20 alleles. The group of alleles that showed a greater increase in binding affinity in mutant peptides than in wild type peptides.

Top 20 alleles			
DRB1*01:04	DRB1*01:20	DRB1*01:02	DRB1*01:21
DRB1*01:26	DRB1*01:10	DRB1*01:23	DRB1*01:09
DRB5*02:05	DRB1*01:31	DRB1*01:01	DRB1*01:05
DRB1*01:07	DRB1*01:08	DRB1*01:12	DRB1*01:19
DRB1*01:22	DRB1*01:25	DRB1*01:27	DRB1*01:30

the top 20 alleles, in which it is possible to highlight 3 distinct clusters. The cluster of non-binder peptides consists of a small percentage of points, highlighting some outliers, whose binding affinity is higher in *wild type* peptides. Using the zoom in tool to focus the plot only on peptides classified as weak binders and strong binders (**Figure 20 B**) demonstrates that the strong binding peptides represent the largest slice of observations, forming a strong and consistent cluster of points where the presence of the mutant amino acid induces changes in the binding affinities for the vast majority of the HLA class II alleles tested.

The second graph, analysing only the peptides predicted as a strong binding and comparing the differences between the *wild type* and mutant peptides, is shown in the **Figure 21**. The limit for an epitope to be considered strong binding peptide was set at 50 nM for binding affinity, identified by the orange line. The trend of observations noted above is strongly reinforced in this graph, bearing in mind that the vast majority of points whose affinity value is less than 50 nM corresponds to mutant peptides. These results strongly suggest that the presence of the R233L mutation in the AccD2 protein may play a key role in the induction of CD4$^+$ T cell mediated the immune responses.

The further detailed analysis performed by HABIT (**Table 3**) was also very informative. The first two columns, labelled "Wild type" and "Mutant", show that the peptides that include the mutation have a significant ($p<0.05$) increase in the number of peptides classified as strong binders and an also a significant ($p<0.05$) decrease in the mean value of binding affinity relative to the corresponding value of the *wild type* peptides.

Based on the percentage of "Predicted SB epitopes", 78 alleles were counted in which the number of strong binders increases in the mutant peptides, directing future validation studies to these alleles. The DRB1*01:04 allele presented the highest percentage of strong binding epitopes, with an associated value of 57.6%, an increase in the number of strong binders greater than 50%. Furthermore, the other class II HLA alleles to be highlighted were: HLA-DRB1*01:20, HLA-DRB1*01:02, HLA-DRB1*01:21, HLA-DRB1*01:25, HLA-DRB1 *01:10 and HLA-DRB1*01:23.

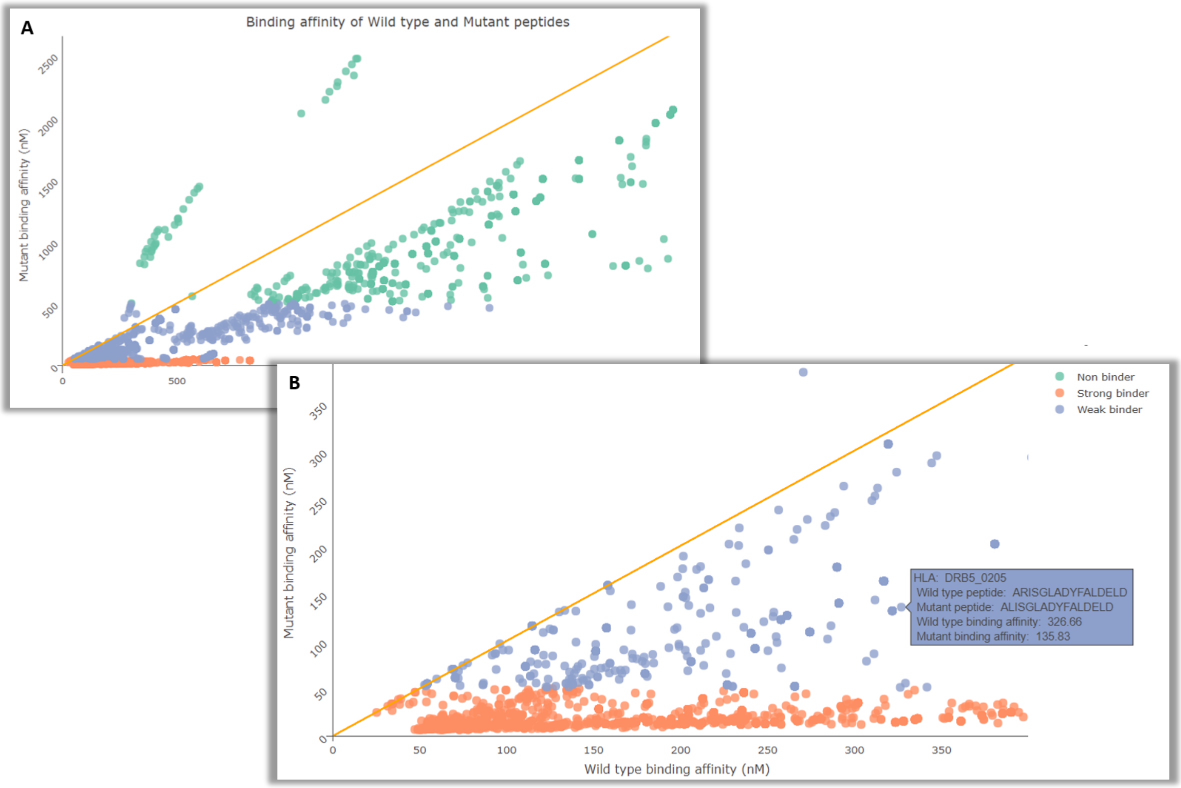


Figure 20: Interactive plot that compares the % rank values of each peptide pair (wild type/mutant) belonging to the AccD2 protein when tested for an HLA molecule. Considering that the greater the distance from an observation to the orange line the larger the difference between the pairs of peptides. **A.** The plot includes all observations of the 20 alleles which showed higher binding affinity to the mutant peptides. **B.** The graph obtained by using the zoom in tool includes only the points of greatest interest to the study of the accd2 protein.

Using this immunoinformatic approach, it was possible to verify that the R233L mutation has a great impact on the number of class II HLA alleles capable of binding to epitopes with high affinity. Considering that HLA molecules have different frequencies throughout the world the response to the epitopes will change depending on the human population being considered. Thus, HABIT also allows to analyse the population coverage of the alleles highlighted as having a significantly increased biding capacity to the mutant AccD2 epitope. The analysis performed by HABIT allowed to focus on 14 class II HLA alleles and presented their population coverage in 15 different geographic regions of the globe (**Table** 4).

Overall, our results suggest that the polymorphism found in the AccD2 protein leads to the presence of epitopes that are predicted to bind strongly to some class II HLA molecules, contrarily to what is observed in the wild type sequence. The population coverage analysis suggests that the mutant peptide might be increasingly recognized by the host populations

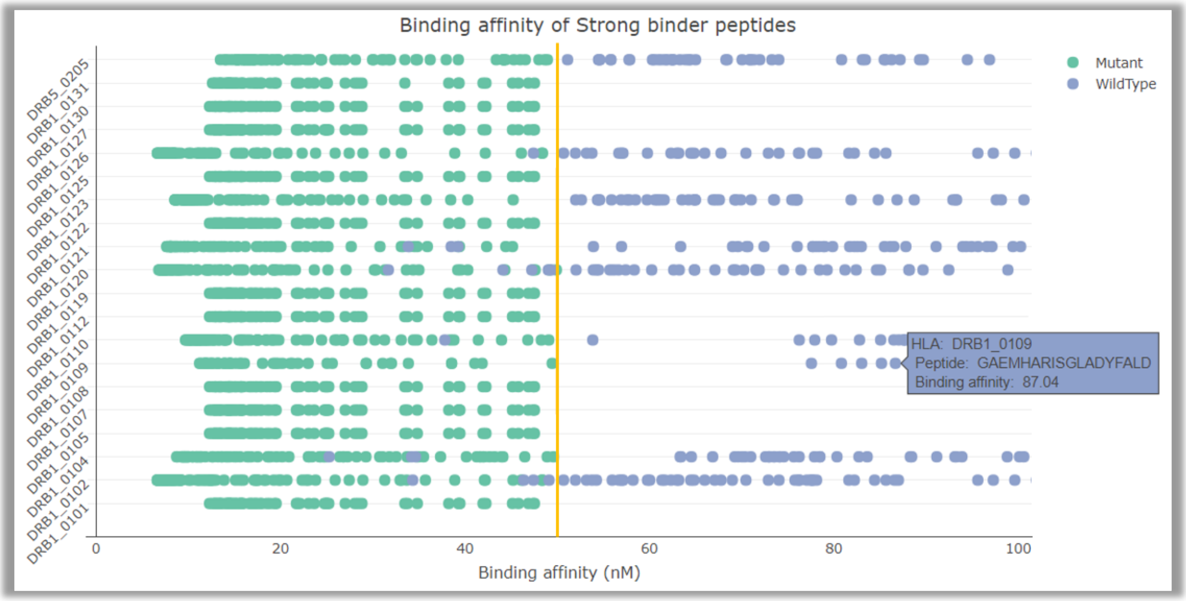
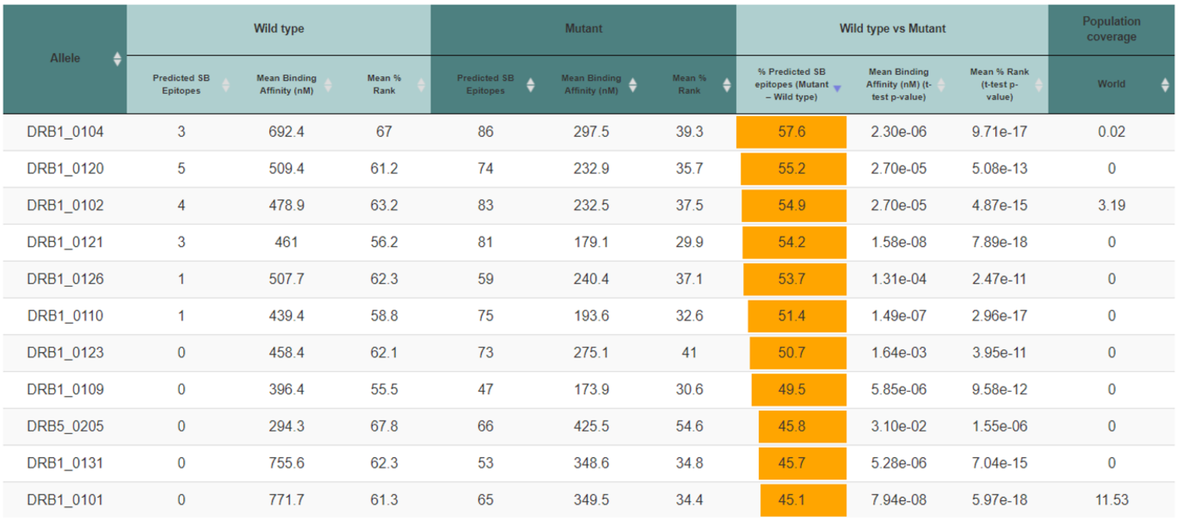


Figure 21: Interactive dot plot that selects all pairs of peptides (wild type/mutant) belonging to the AccD2 that have at least one strong binding peptide. By allele, this plot allows comparison of the % rank values of two peptides and identification of similarity between them. The orange vertical line overcomes the limit of stong binding peptides.

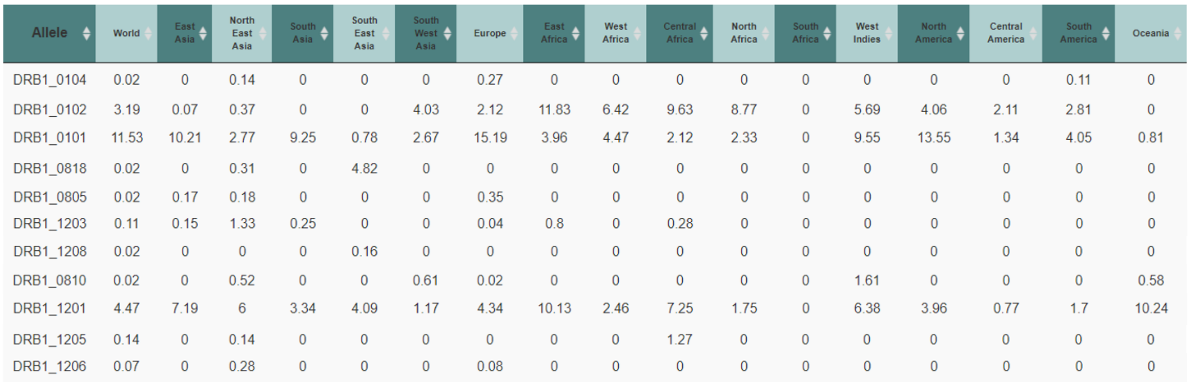
Table 3: Individual and comparative analysis of the outputs obtained from the epitope prediction tools for the wild type and mutant sequences. The results were calculated for each selected class II allele and sorted, in the table, in decreasing order of column "% Predicted epitopes (mutant - wild type)".



Allele	Wild type			Mutant			Wild type vs Mutant			Population coverage
	Predicted SB Epitopes	Mean Binding Affinity (nM)	Mean % Rank	Predicted SB Epitopes	Mean Binding Affinity (nM)	Mean % Rank	% Predicted SB epitopes (Mutant – Wild type)	Mean Binding Affinity (nM) (t-test p-value)	Mean % Rank (t-test p-value)	World
DRB1_0104	3	692.4	67	86	297.5	39.3	57.6	2.30e-06	9.71e-17	0.02
DRB1_0120	5	509.4	61.2	74	232.9	35.7	55.2	2.70e-05	5.08e-13	0
DRB1_0102	4	478.9	63.2	83	232.5	37.5	54.9	2.70e-05	4.87e-15	3.19
DRB1_0121	3	461	56.2	81	179.1	29.9	54.2	1.58e-08	7.89e-18	0
DRB1_0126	1	507.7	62.3	59	240.4	37.1	53.7	1.31e-04	2.47e-11	0
DRB1_0110	1	439.4	58.8	75	193.6	32.6	51.4	1.49e-07	2.96e-17	0
DRB1_0123	0	458.4	62.1	73	275.1	41	50.7	1.64e-03	3.95e-11	0
DRB1_0109	0	396.4	55.5	47	173.9	30.6	49.5	5.85e-06	9.58e-12	0
DRB5_0205	0	294.3	67.8	66	425.5	54.6	45.8	3.10e-02	1.55e-06	0
DRB1_0131	0	755.6	62.3	53	348.6	34.8	45.7	5.28e-06	7.04e-15	0
DRB1_0101	0	771.7	61.3	65	349.5	34.4	45.1	7.94e-08	5.97e-18	11.53

in the regions of Europe, East Asia, East Africa, Central Africa, West India and North America.

Table 4: The frequencies of each allele in 15 geographic regions.



Allele	World	East Asia	North East Asia	South Asia	South East Asia	South West Asia	Europe	East Africa	West Africa	Central Africa	North Africa	South Africa	West Indies	North America	Central America	South America	Oceania
DRB1_0104	0.02	0	0.14	0	0	0	0.27	0	0	0	0	0	0	0	0	0.11	0
DRB1_0102	3.19	0.07	0.37	0	0	4.03	2.12	11.83	6.42	9.63	8.77	0	5.69	4.06	2.11	2.81	0
DRB1_0101	11.53	10.21	2.77	9.25	0.78	2.67	15.19	3.96	4.47	2.12	2.33	0	9.55	13.55	1.34	4.05	0.81
DRB1_0818	0.02	0	0.31	0	4.82	0	0	0	0	0	0	0	0	0	0	0	0
DRB1_0805	0.02	0.17	0.18	0	0	0	0.35	0	0	0	0	0	0	0	0	0	0
DRB1_1203	0.11	0.15	1.33	0.25	0	0	0.04	0.8	0	0.28	0	0	0	0	0	0	0
DRB1_1208	0.02	0	0	0	0.16	0	0	0	0	0	0	0	0	0	0	0	0
DRB1_0810	0.02	0	0.52	0	0	0.61	0.02	0	0	0	0	0	1.61	0	0	0	0.58
DRB1_1201	4.47	7.19	6	3.34	4.09	1.17	4.34	10.13	2.46	7.25	1.75	0	6.38	3.96	0.77	1.7	10.24
DRB1_1205	0.14	0	0.14	0	0	0	0	0	0	1.27	0	0	0	0	0	0	0
DRB1_1206	0.07	0	0.28	0	0	0	0.08	0	0	0	0	0	0	0	0	0	0

4.2.3 *Discussion*

The case study herein presented was a detailed *in silico* characterisation based on HABIT of the HLA binding properties of a variable *M. tuberculosis* candidate epitope. This step was intended to test the behaviour and usefulness of HABIT in the analysis of real data, in the context of an ongoing research project. Furthermore, it was also used to evaluate its ability to respond promptly to the main issues raised in the study of antigenic epitopes determined by prediction algorithms. Thus, the AccD2 protein was submitted to HABIT and the results obtained were analysed from the scientific point of view, considering the information consistency and the impact of the verified differences between wild type and mutant sequences.

By the use of the user friendly and quick interface of HABIT, we were able to analyse a large amount of data and display it in interactive and customizable graphs and tables. In a single web server, an automatic and robust pipeline is implemented that allows the analysis of the impact of an amino acid variant on T cell antigenicity, based on the comparison of the individual predictions of the *wild type* and mutant sequences, the significance of statistical tests, and the information on the distribution of alleles across the globe. This implementation made it possible to perform an integrated analysis of HLA-epitope binding affinity prediction, without the need for programming skills to analyse the data.

Most interestingly, the results obtained for the impact of the R233L mutation in the binding of AccD2 peptides to class II HLA molecules was very promising. It showed that the presence of this mutation is responsible for inducing generalized and significant changes in the binding affinity to class II HLA molecules but not to class I HLA molecules. This highlights a novel $CD4^+$ T cell epitope that is present in a subset of *M. tuberculosis* bacteria with a high potential to influence host immune responses. This is of relevance due to the relevant role that $CD4^+$ T cells have in the immunity against tuberculosis (Mogues et al.,

2001) and might be important for the development of vaccines to replace or complement the efficacy of BCG.

In silico analyses with HABIT also heighted that the impact of these mutation might not be the same in all world regions due to the differences in frequency of HLA alleles in the human populations. Nonetheless, it is important to refer that the mutation was predicted to have a large impact on the recognition by the class II HLA-DRB1*01:01, a molecule that exhibits a wide global geographical distribution.

TB is the infectious disease with higher mortality in the world. An effective vaccine would be key for the control of this disease but its development requires advances in our understanding of how *M. tuberculosis* evolved to modulate the human immune response. While most pathogens have evolved to evade host immunity by antigenic variation, *M. tuberculosis* seems to apply a different strategy. It has been shown that known human T cell epitopes are evolutionarily hyperconserved, with the large majority of the known T cell epitopes analysed showing no amino acid variants (Coscolla et al., 2015). *M. tuberculosis* possibly uses these hyperconserved epitopes to exploit the induced host T cell responses in its benefit to ensure the tissue damage necessary to potentiate transmission (Rodrigo et al., 1997). Nonetheless, some antigens present amino acid variants (Coscolla et al., 2015; Yruela et al., 2016). This is important, as the alteration of a single amino acid in an epitope can impact its affinity to a specific HLA molecule, thus influencing the T cell synapse and modulating the level and type of immune response elicited (da Silva, 2003; Copin et al., 2014). The analysis performed using HABIT highlights that AccD2 harbours one of these variable T cell epitope candidates. The results obtained in this work reinforce that the study of the immunogenic properties of this specific mutation (R233L) might be relevant for the future development of vaccines and also immune response-based diagnostic test.

5

CONCLUSION

This last chapter is devoted to the main conclusions retained throughout the different stages of development of the present thesis, as well as a brief reference to limitations and ideas to be included in the future work.

5.1 OVERVIEW

The technological advances in genome sequencing platforms provoked a relevant change in the way biomedical research is conducted. It is now possible to have an in depth and continuous knowledge of the genetic variations in different genotypes of pathogens or cancer tissues. Combining this information with powerful computational tools and algorithms will have a decisive impact in the development of immunotherapy strategies that harbour the potential to solve the most relevant health problems in the world, such as cancer or TB. It will be possible to predict with accuracy and in a personalized way the impact that variants have and which antigens are the most capable of developing beneficial immune responses. These systems biology-based strategies are starting to produce results on the pipelines of vaccine design, reducing the time and money needed to identify the epitope vaccine candidates (Rappuoli, 2001; Kanampalliwar et al., 2013).

Most importantly, a large number of algorithms and tools have been developed in the last decade largely augmenting the capacity to predict epitopes with higher binding affinity to HLA molecules. However, existing approaches do not include integrated analyses and data treatment at different levels. In general, these tools require programming skills for successful use. On the one hand, most tools require local installation of programs that are command-line based and generate large outputs of information in raw files that need to be analysed.

Due to the present and future relevance of immunotherapy, the ability to interpret and understand the results generated by immunoinformatics tools, clarifying its potential biological meaning in the immunological context, is a very valuable asset. Thus, the development of this thesis had as main objective to implement a web server capable of responding effectively to the needs of the researchers that focus on the impact of genetic variants in predicted immune responses. HABIT functions as a resource that combines and integrates epitope prediction tools, to provide a detailed and comparative analysis of the influence of a given amino acid variant on binding affinity to HLA-peptides. The results are presented to the user through a simple and intuitive interface, allowing interactive data analysis according to the criteria that best fit the study.

During the development of this web server, several challenges have emerged, as well as the need to implement approaches totally different from what was idealized in an initial phase. One of these challenges that was overcome concerned the process of integrating the epitope prediction tools, and the consequent development of a parallel system to the shiny-based server module. Taking into account that these are fully independent programs, it was mandatory to configure each individual tool, to ensure that they did not interfere with the operation and sessions of the server.

The case study applied to the *Mycobacterium tuberculosis* AccD2 protein and in particular to the amino acid variant R233L, allowed to test the resources available in HABIT and to make a critical evaluation of the efficiency and relevance of the obtained results. Thus, integrating and exposing the data through an interactive graphical interface, makes the web server a powerful approach with enormous potential for understanding and streamlining the fundamental processes of immunotherapy, emphasizing the need for a tool capable of integrating and processing data from different sources.

Overall, HABIT provides a fast and user-friendly way to generate and analyse large sets of data to answer the following questions: “What is the predicted impact of an amino acid mutation in HLA-binding?”; “Are the differences statistically significant?” and “What percentage/subset of the human population is more likely to respond to the peptides?”. With this, HABIT might advance the discovery of molecular determinants that influence variation in T cell-mediated immune responses, thus contributing to rational immunotherapy design.

5.2 LIMITATIONS

The present project, as well as any scientific work, presents some limitations. The main limitations of HABIT at this stage are:

- The application is designed to accept the input of a single amino acid variant at the time. So, for each submission, the user can only compare the wild type sequence with one mutation. While the analysis of the impact of one mutation is the most typical for TB or cancer researcher, it might be interesting for the researcher to investigate a region where more than one mutation is present. To do this at the moment the user has to perform more than one submission.

- The fact that HABIT integrates two epitopes prediction tools with a fully independent operation, increases the computational time to obtain the outputs and return the results to the users. This limitation is due to the fact that *NetMHCpan* and *NetMHCIIpan*n were programmed by the developers and independent tools.

- The open source *shiny* server provided by R is the platform that hosts HABIT allowing it to be launched online, and thus, as any open source modality has some limitations regarding the available features. Indeed, this version only allows each web application to have an open R session, that is, the web server can be used by one user at a time.

These are the main limitations identified in HABIT and will be the priority for future developments as alluded in the next section.

5.3 FUTURE PERSPECTIVES

The priorities in terms of future development of HABIT are:

- Add non parametric statistical tests, to complement the information provided by the t-tests performed at the mean of the binding affinity and at the mean of the % rank, in cases where the normality between the peptides is not verified.

- Also to improve the statistical analysis performed by the web server, include a correction method for multiple comparisons in the presented t-tests, for example, the Bonferroni method. Considering this a more conservative method, its implementation will aim to ensure the level of confidence of the results obtained by the comparisons between wild type and mutant sequences.

- Develop a method for comparing several amino acid variants in different positions, to compare and visualize the impact of multiple mutations at the same time. This topic would add a great value to the analysis process available by the web server.

- Based on the above method, improve existing plots to include analyses of several mutations simultaneously and implement new interactive plots that show, for example, the impact of each mutation on its position of the protein sequence.

- HABIT is *shiny* app that runs R in a single thread program, which can use only one CPU core, for multiple concurrent requests. The server where HABIT is implemented has 24 cores so a Linux monitoring system will be developed and tested to deal with multiple concurrent users by starting another R process on different CPU cores.

The features implemented in HABIT within the scope of the thesis constitute a significant advance for the development immunoinformatic studies to predict the impact of amino acid variants in T cell-mediated immune responses. Inevitably, there is room for future developments in the tool but all the initially proposed objectives have been accomplished.

BIBLIOGRAPHY

B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular biology of the cell*. Annals of Botany, 2003.

M. Andreatta, E. Karosiene, M. Rasmussen, A. Stryhn, S. Buus, and M. Nielsen. Accurate pan-specific prediction of peptide-mhc class ii binding affinity with improved binding core identification. *Immunogenetics*, 67(11-12):641–650, 2015.

S. Bahram. Complete sequence and gene map of a human major histocompatibility complex. the mhc sequencing consortium. *Nature*, 401:921–923, 1999.

P. Baldi and S. Brunak. *Bioinformatics: the machine learning approach*. MIT press, 2001.

C. Basset, J. Holton, R. OMahony, and I. Roitt. Innate immunity and pathogen–host interaction. *Vaccine*, 21:S12–S23, 2003.

V. Brusic and N. Petrovsky. Immunoinformatics-the new kid in town. In *Novartis Foundation Symposium*, pages 3–22. Chichester; New York; John Wiley; 1999, 2003.

A. Calmette, C. Guérin, A. Boquet, and L. Nègre. *La vaccination préventive contre la tuberculose par le" BCG,"*. Masson et cie, 1927.

P. Cano, W. Klitz, S.J. Mack, M. Maiers, S.G.E. Marsh, H. Noreen, E.F. Reed, D. Senitzer, M. Setterholm, A. Smith, et al. Common and well-documented hla alleles: report of the ad-hoc committee of the american society for histocompatiblity and immunogenetics. *Human immunology*, 68(5):392–417, 2007.

R.R. Caspi. Immunotherapy of autoimmunity and cancer: the penalty for success. *Nature Reviews Immunology*, 8(12):970–976, 2008.

W. Chang, J. Cheng, and J.J. Allaire. Shiny: Web application framework for r. 2017.

F.A. Chaves, A.H. Lee, J.L. Nayak, K.A. Richards, and A.J. Sant. The utility and limitations of current web-available algorithms to predict peptides recognized by cd4 t cells in response to pathogen infection. *The Journal of Immunology*, 188(9):4235–4248, 2012.

M.D. Cooper. The early history of b cells. *Nature Reviews Immunology*, 15(3):191–197, 2015.

R. Copin, M. Coscollá, E. Efstathiadis, S. Gagneux, and J.D. Ernst. Impact of in vitro evolution on antigenic diversity of mycobacterium bovis bacillus calmette-guerin (bcg). *Vaccine*, 32(45):5998–6004, 2014.

M. Coscolla, R. Copin, J. Sutherland, F. Gehre, B. de Jong, O. Owolabi, G. Mbayo, F. Giardina, J.D. Ernst, and S. Gagneux. M. tuberculosis t cell epitope analysis reveals paucity of antigenic variation and identifies rare variable tb antigens. *Cell host & microbe*, 18(5): 538–548, 2015.

J. da Silva. The evolutionary adaptation of hiv-1 to specific immunity. *Current HIV research*, 1(3):363–371, 2003.

J. Dausset. Iso-leuco-anticorps. *Acta haematologica*, 20(1-4):156–166, 1958.

J. Dausset. The major histocompatibility complex in man: Past, present and future concepts. *Science*, 213(4515):1469–1474, 1981.

P.J. Delves and I.M. Roitt. The immune system. *New England Journal of Medicine*, 343(1): 37–49, 2000.

R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification*. New York: Wiley, 2001.

M.T. Ehebauer, M. Zimmermann, A.J. Jakobi, E.E. Noens, D. Laubitz, B. Cichocki, H. Marrakchi, M. Lanéelle, M. Daffé, and C. Sachse. Characterization of the mycobacterial acyl-coa carboxylase holo complexes reveals their functional expansion into amino acid catabolism. *PLoS pathogens*, 11(2):e1004623, 2015.

J.D. Ernst. The immunological life cycle of tuberculosis. *Nature Reviews Immunology*, 12(8), 2012.

K. Falk, O. Rotzschke, S. Stevanovic, G. Jung, and H.G. Rammensee. Allele-specific motifs revealed by sequencing of self-peptides eluted from mhc molecules. *Nature*, 351:290–296, 1991.

D. Farrell and S.V. Gordon. Epitopemap: a web application for integrated whole proteome epitope prediction. *BMC bioinformatics*, 16(1):221, 2015.

R. Gande, K.J.C Gibson, A.K. Brown, K. Krumbach, L.G. Dover, H. Sahm, S. Shioyama, T. Oikawa, G.S Besra, and L. Eggeling. Acyl-coa carboxylases (accd2 and accd3), together with a unique polyketide synthase (cg-pks), are key to mycolic acid biosynthesis in corynebacterianeae such as corynebacterium glutamicum and mycobacterium tuberculosis. *Journal of Biological Chemistry*, 279(43):44847–44857, 2004.

R.N. Germain. Mhc-dependent antigen processing and peptide presentation: providing ligands for t lymphocyte activation. *Cell*, 76(2):287–299, 1994.

S. Henikoff and J.G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

C.A. Janeway, P. Travers, M. Walport, and M.J. Shlomchik. *Immunobiology: the immune system in health and disease*, volume 1. Current Biology, 1997.

C.W. Kanaly, D. Ding, A.B. Heimberger, and J.H. Sampson. Clinical applications of a peptide-based vaccine for glioblastoma. *Neurosurgery Clinics of North America*, 21(1):95–109, 2010.

A.M. Kanampalliwar, S. Rajkumar, A. Girdhar, and T. Archana. Reverse vaccinology: basics and applications. *J Vaccines Vaccin*, 4(6):194–198, 2013.

S.H.E. Kaufmann. Envisioning future strategies for vaccination against tuberculosis. *Nature reviews. Immunology*, 6(9):699, 2006.

S.H.E. Kaufmann. Future vaccination strategies against tuberculosis: thinking outside the box. *Immunity*, 33(4):567–577, 2010.

Y. Kim, J. Ponomarenko, Z. Zhu, D. Tamang, P. Wang, J. Greenbaum, C. Lundegaard, A. Sette, O. Lund, P.E. Bourne, et al. Immune epitope database analysis resource. *Nucleic acids research*, page gks438, 2012.

J. Klein. Seeds of time: fifty years ago peter a. gorer discovered the h-2 complex. *Immunogenetics*, 24(6):331–338, 1986.

R. Koch. The etiology of tuberculosis. *Reviews of infectious diseases*, 4(6):1270–1274, 1982.

Y.M. Kong, J.C. Flynn, J.P. Banga, and C.S. David. Application of hla class ii transgenic mice to study autoimmune regulation. *Thyroid*, 17(10):995–1003, 2007.

S. Kreiter, M. Vormehr, N. van de Roemer, M. Diken, M. Löwer, J. Diekmann, S. Boegel, B. Schrörs, F. Vascotto, J.C. Castle, et al. Mutant mhc class ii epitopes drive therapeutic immune responses to cancer. *Nature*, 520(7549):692–696, 2015.

S. Kumar and D. Chandra. A therapeutic perspective of cytokines in tumor management. *Inflammation and Cell Signaling*, 1(2), 2014.

W. Li, M.D. Joshi, S. Singhania, K.H. Ramsey, and A.K. Murthy. Peptide vaccine: progress and challenges. *Vaccines*, 2(3):515–536, 2014.

A.M. Little and P. Parham. Polymorphism and evolution of hla class i and ii genes and molecules. *Reviews in immunogenetics*, 1(1):105–123, 1999.

M. Lombard, P.P. Pastoret, and A.M Moulin. A brief history of vaccines and vaccination. *Revue Scientifique et Technique-Office International des Epizooties*, 26(1):29–48, 2007.

P.S. Maclin, J. Dempsey, J. Brooks, and J. Rand. Using neural networks to diagnose cancer. *Journal of medical systems*, 15(1):11–19, 1991.

S.G.E. Marsh, E.D. Albert, W.F. Bodmer, R.E. Bontrop, B. Dupont, H.A. Erlich, M. Fernandez-Vina, D.E. Geraghty, R. Holdsworth, C.K. Hurley, et al. Nomenclature for factors of the hla system, 2010. *Tissue antigens*, 75(4):291, 2010.

W.S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

R. Medzhitov and C. Janeway Jr. Innate immunity. *N Engl J Med*, 2000(343):338–344, 2000.

T. Mitchell. *Machine Learning*. New York: McGraw Hill, 1997.

T.H. Mogensen. Pathogen recognition and inflammatory signaling in innate immune defenses. *Clinical microbiology reviews*, 22(2):240–273, 2009.

T. Mogues, M.E. Goodrich, L. Ryan, R. LaCourse, and R.J. North. The relative importance of t cell subsets in immunity and immunopathology of airborne mycobacterium tuberculosis infection in mice. *Journal of Experimental Medicine*, 193(3):271–280, 2001.

A.J. Mungall, S.A. Palmer, S.K. Sims, C.A. Edwards, J.L. Ashurst, et al. The dna sequence and analysis of human chromosome 6. *Nature*, 425(6960):805, 2003.

M. Nielsen and M. Andreatta. Netmhcpan-3.0; improved prediction of binding to mhc class i molecules integrating information from multiple receptor and peptide length datasets. *Genome medicine*, 8(1):33, 2016.

M. Nielsen, C. Lundegaard, P. Worning, S.L. Lauemøller, K. Lamberth, S. Buus, S. Brunak, and O. Lund. Reliable prediction of t-cell epitopes using neural networks with novel sequence representations. *Protein Science*, 12(5):1007–1017, 2003.

M. Nielsen, C. Lundegaard, T. Blicher, K. Lamberth, M. Harndahl, S. Justesen, G. Røder, B. Peters, A. Sette, O. Lund, et al. Netmhcpan, a method for quantitative predictions of peptide binding to any hla-a and-b locus protein of known sequence. *PloS one*, 2(8):e796, 2007.

M. Nielsen, S. Justesen, O. Lund, C. Lundegaard, and S. Buus. Netmhciipan-2.0-improved pan-specific hla-dr predictions using a novel concurrent alignment and weight optimization training procedure. *Immunome research*, 6(1):9, 2010.

A. O'Garra, P.S.Redford, F.W. McNab, C.I. Bloom, R.J. Wilkinson, and M.P.R. Berry. The immune response in tuberculosis. *Annual review of immunology*, 31:475–527, 2013.

N.S. Osório, F. Rodrigues, S. Gagneux, J. Pedrosa, M. Pinto-Carbó, A.G. Castro, D. Young, I. Comas, and M. Saraiva. Evidence for diversifying selection in a set of mycobacterium tuberculosis genes in response to antibiotic-and nonantibiotic-related pressure. *Molecular biology and evolution*, 30(6):1326–1336, 2013.

P. Oyarzun and B. Kobe. Computer-aided design of t-cell epitope-based vaccines: addressing population coverage. *International journal of immunogenetics*, 42(5):313–321, 2015.

P. Oyarzún, J.J. Ellis, M. Bodén, and B. Kobe. Predivac: Cd4+ t-cell epitope prediction for vaccine design that covers 95% of hla class ii dr protein diversity. *BMC bioinformatics*, 14 (1):52, 2013.

J. Parkin and B. Cohen. An overview of the immune system. *The Lancet*, 357(9270):1777–1789, 2001.

D.W. Parsons, S. Jones, X. Zhang, J.C. Lin, R.J. Leary, P. Angenendt, P. Mankoo, H. Carter, I. Siu, G.L. Gallia, et al. An integrated genomic analysis of human glioblastoma multiforme. *Science*, 321(5897):1807–1812, 2008.

A. Patronov and I. Doytchinova. T-cell epitope vaccine design by immunoinformatics. *Open biology*, 3(1):120139, 2013.

Y. Perrie, D. Kirby, V.W. Bramwell, and A.R. Mohammed. Recent developments in particulate-based vaccines. *Recent patents on drug delivery & formulation*, 1(2):117–129, 2007.

R. Rappuoli. Reverse vaccinology, a genome-based approach to vaccine development. *Vaccine*, 19(17):2688–2691, 2001.

S. Riedel. Edward jenner and the history of smallpox and vaccination. *Proceedings (Baylor University. Medical Center)*, 18(1):21, 2005.

T. Rodrigo, J.A. Cayla, P. Garcia de Olalla, H. Galdós-Tangüis, J.M. Jansà, P. Miranda, and T. Brugal. Characteristics of tuberculosis patients who generate secondary cases. *The International Journal of Tuberculosis and Lung Disease*, 1(4):352–357, 1997.

D.S. Rosa, S.P. Ribeiro, and E. Cunha-Neto. Cd4$^+$ t cell epitope discovery and rational vaccine design. *Archivum immunologiae et therapiae experimentalis*, 58(2):121–130, 2010.

D.E. Rumelhart, J.L. McClelland, and G.E. Hinton. The appeal of parallel distributed processing. *MIT Press, Cambridge MA*, pages 3–44, 1986.

T.N. Schumacher and R.D. Schreiber. Neoantigens in cancer immunotherapy. *Science*, 348 (6230):69–74, 2015.

A. Sette and R. Rappuoli. Reverse vaccinology: developing vaccines in the era of genomics. *Immunity*, 33(4):530–541, 2010.

R.J. Simes. Treatment selection for cancer patients: application of statistical decision theory to the treatment of advanced ovarian cancer. *Journal of chronic diseases*, 38(2):171–186, 1985.

G.D. Snell. Methods for the study of histocompatibility genes. *J Genet*, 49(2):87–108, 1948.

O. Takeuchi and S. Akira. Pattern recognition receptors and inflammation. *Cell*, 140(6): 805–820, 2010.

T. Trolle, I.G. Metushi, J.A. Greenbaum, Y. Kim, J. Sidney, O. Lund, A. Sette, B. Peters, and M. Nielsen. Automated benchmarking of peptide-mhc class i binding predictions. *Bioinformatics*, page btv123, 2015.

J. Trowsdale and J.C. Knight. Major histocompatibility complex genomics and human disease. *Annual review of genomics and human genetics*, 14:301–323, 2013.

P.J. van den Elsen, S.J.P. Gobin, M.C.J.A Van Eggermond, and A. Peijnenburg. Regulation of mhc class i and ii gene transcription: differences and similarities. *Immunogenetics*, 48 (3):208–221, 1998.

R. Vita, J.A. Overton, J.A. Greenbaum, J. Ponomarenko, J.D. Clark, J.R. Cantrell, D.K. Wheeler, J.L. Gabbard, D. Hix, A. Sette, et al. The immune epitope database (iedb) 3.0. *Nucleic acids research*, 43(D1):D405–D412, 2015.

I.F. Voutsas, A.D. Gritzapis, L.G. Mahaira, M. Salagianni, E. Hofe, N.L. Kallinteris, and C.N. Baxevanis. Induction of potent cd4+ t cell-mediated antitumor responses by a helper her-2/neu peptide linked to the ii-key moiety of the invariant chain. *International journal of cancer*, 121(9):2031–2041, 2007.

WHO. Global tuberculosis report. Technical report, Geneva: WHO, 2016.

Z.R. Yang, R. Thomson, P. McNeil, and R.M. Esnouf. Ronn: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. *Bioinformatics*, 21(16):3369–3376, 2005.

I. Yruela, B. Contreras-Moreira, C. Magalhães, N.S. Osório, and J. Gonzalo-Asensio. Mycobacterium tuberculosis complex exhibits lineage-specific variations affecting protein ductility and epitope recognition. *Genome biology and evolution*, 8(12):3751–3764, 2016.

L. Zhang, Y. Chen, H. Wong, S. Zhou, H. Mamitsuka, and S. Zhu. Tepitopepan: extending tepitope for peptide binding prediction covering over 700 hla-dr molecules. *PLoS One*, 7 (2):e30483, 2012a.

L. Zhang, K. Udaka, H. Mamitsuka, and S. Zhu. Toward more accurate pan-specific mhc-peptide binding prediction: a review of current methods and tools. *Briefings in bioinformatics*, 13(3):350–364, 2012b.

R.M. Zinkernagel and P.C. Doherty. Restriction of in vitro t cell-mediated cytotoxicity in lymphocytic choriomeningitis within a syngeneic or semiallogeneic system nature. 1974.