

2024

In silico prediction of C57BL/6 mouse T-cell epitopes: enhancing immunogenicity assessment with PREDBL6

<https://hdl.handle.net/2144/50615>

"Downloaded from OpenBU. Boston University's institutional repository."

BOSTON UNIVERSITY
METROPOLITAN COLLEGE

Thesis

**IN SILICO PREDICTION OF C57BL/6 MOUSE T-CELL EPITOPES:
ENHANCING IMMUNOGENICITY ASSESSMENT WITH PREDBL6**

by

ZI TIAN ZHEN

B.S., University of British Columbia, 2021

Submitted in partial fulfillment of the
requirements for the degree of
Master of Science

2024

Approved by

First Reader

Guanglan Zhang, Ph.D.
Associate Professor and Chair of Computer Science

Second Reader

Loubomir T. Chitkushev, Ph.D.
Senior Associate Dean for Academic Affairs
Associate Professor of Computer Science

Third Reader

Derin B. Keskin, Ph.D.
Lead Immunologist
Translational Immuno-Genomics Lab
Dana-Farber Harvard Cancer Institute

Adjunct Associate Professor of Computer Science
Boston University, Metropolitan College

ACKNOWLEDGMENTS

I was fortunate to receive invaluable assistance from numerous individuals throughout the course of this project. I want to express my deepest appreciation to my advisor, Guanglan Zhang, whose unwavering support and guidance have been instrumental throughout this project. Her mentorship, provided tirelessly day and night, helped me to expand my knowledge within this field and was crucial in making this project possible. Dr. Derin Keskin has provided invaluable insight and experimental data for this project. I have greatly benefited from Derin's extensive experience in planning and managing an immunological project. I also want to thank Yuhe Wang for her assistance in establishing the structure of our neural network model. Collaborating with her was a pleasure, and together, we successfully published our preliminary results in the CSECS conference. Finally, I would like to gratefully acknowledge Guancheng Huang for his role in developing the web application for our PREDDBL6, and Yi Zeng for their collaboration in deploying the application to the BU MET server.

**IN SILICO PREDICTION OF C57BL/6 MOUSE T-CELL EPITOPES:
ENHANCING IMMUNOGENICITY ASSESSMENT WITH PREDBL6**

ZI TIAN ZHEN

ABSTRACT

The MHC class I antigen processing pathway involves multiple steps: 1) the proteasome selectively cleaves intracellular proteins into short peptides, typically 8-11 amino acids long; 2) TAPs (transporter associated with antigen processing) selectively transport some of these peptides to the endoplasmic reticulum (ER); 3) aminopeptidases may further degrade the peptides in the ER; 4) some of the peptides bind MHC molecules, and finally; 5) peptide-MHC complexes are transported to the cell surface for recognition by CD8⁺ T cells. Peptides presented by MHC and recognized by T cells are called T-cell epitopes. MHC class I restricted T cell epitopes play a crucial role in the immune surveillance of intracellular pathogens.

As MHC binding is considered the most selective step in T cell recognition, many existing bioinformatics systems focus on modeling this step to predict MHC binders. However, modeling MHC binding alone is insufficient for accurate immunogenicity predictions, often resulting in false positives. With the recent technological advancements, large amounts of mass spectrometry (MS)-identified MHC class I ligands became available to the public, making it possible to incorporate information from antigen processing steps before MHC binding. We collected >5,000 binding peptides and >4,000 eluted ligands for H2-D^b, and >5,000 binding peptides and >5,000 eluted ligands for H2-K^b.

The thermostability assessment of MHC peptide binding evaluates the strength and duration of the interaction between the peptide and the MHC molecule under varying temperatures. Studies have shown a positive correlation between thermostability and immunogenicity, as the stability of the peptide-MHC complexes affects the efficiency of antigen presentation and the downstream activation of T cells. Higher thermostability of peptide-MHC complexes allows for more extended interactions with T cell receptors, thus a higher likelihood of T cell activation. Our collaborators at the Dana-Farber Cancer Institute performed temperature gradient experiments to investigate the stability of peptide-MHC complexes for H2-D^b and H2-K^b alleles under three temperature conditions, 37°C, 50°C, and 70°C, using the MS technique. The binding peptides were isolated using immunoprecipitation (IP) techniques. Peptides with lower binding stability tended to dissociate from the MHC molecules as the temperature increased, indicating reduced binding stability on the MHC surface. We ended up with over 3,000 H2-D^b binding peptides and over 5,000 H2-K^b binding peptides. The data enable us to perform a comprehensive thermostability analysis of MHC binding.

In this thesis project, we developed a computational system for identifying T-cell epitopes in C57BL mice by integrating relevant contributing factors, such as the antigen processing steps before MHC binding and thermostability, with the MHC binding predictions. Utilizing deep learning methods, we first trained and rigorously validated the binding prediction models using naturally eluted H2-K^b and H2-D^b ligands collected from public resources. Then, we built Thermostability models using proprietary data generated by our collaborators. We compared the performance of our models with that of

NetMHCPan-4.1, an online prediction tool validated by many benchmark studies to be one of the most accurate predictors. Our integrated model, combining the binding and the Thermostability models, exhibited superior predictive capabilities using an external validation dataset, surpassing the overall performance of the NetMHCPan-4.1 model.

We consolidated the models into a user-friendly web-based application named PREDBL6 to facilitate accurate predictions of immunogenic peptides that stably bind H2b molecules and stimulate immune responses in C57BL/6 mice. To our knowledge, this is the first online T cell epitope prediction system that simulates MHC binding and considers other antigen processing steps and thermostability in a model organism.

PREDBL6 is available at <http://met-hilab.org:3001/tool>

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iv
ABSTRACT	v
TABLE OF CONTENTS.....	viii
LIST OF TABLES	xi
LIST OF FIGURES	xii
LIST OF ABBREVIATIONS.....	xiv
1. INTRODUCTION	1
1.1. T-cells, MHC Molecules, and Epitope-based Vaccines	1
1.2. Reverse Vaccinology	2
1.3. Data Type (BA vs EL)	4
1.4. Thesis Statement	6
1.5. Contribution of the Thesis	6
1.6. Organization of the Thesis	7
2. LITERATURE REVIEW	9
2.1. Peptide-based T cell Vaccines	9
2.2 Existing Computational Systems	11
2.3. IEDB Database	15
3. MATERIALS AND METHODS.....	16
3.1. Data Collection	16
3.1.1 The Collection of Eluted Ligands and MHC Binding Peptides.....	16
3.1.2 Thermostability Data from Dana Farber Cancer Institute	18

3.2 Data Transformation	20
3.2.1 One-Hot-Encoding	20
3.2.2 BLOSUM Encoding	21
3.3 Machine Learning Models	22
3.4 Study Design	25
3.5 Validation Dataset.....	29
3.6 Evaluation Metrics	30
3.6.1 ROC & AUC	30
3.6.2 K-fold Cross Validation	31
3.7 Implementation	32
4. RESULTS	34
4.1 Comparative Analysis of Machine Learning Models	34
4.2 MHC Motifs.....	37
4.3 Internal Evaluation of EL Models	41
4.4 Finding the Thresholds for EL models	42
4.5 Internal Evaluation of Thermal Stability Models	43
4.6 External Validation	45
4.7 Implementation of PREDDBL6	48
4.7.1 Backend.....	48
4.7.2 Frontend.....	49
4.7.3 Implementation	50
4.8 Webpage	51

5. DISCUSSION.....	55
5.1 EL+TS Models	55
5.2 Webpage	58
5.3 Online Tool Maintenance	58
6. CONTRIBUTION AND FUTURE WORK.....	60
7. BIBLIOGRAPHY.....	62
CURRICULUM VITAE.....	67

LIST OF TABLES

Table 3.1 Eluted H2-D ^b and H2-K ^b ligands collected from IEDB and NetMHCpan-4.1 dataset.	17
Table 3.2 Additional BA data for H2-D ^b and H2-K ^b alleles collected from IEDB.	18
Table 3.3 Peptide counts of the H2-D ^b and H2-K ^b stability dataset across various lengths and temperature conditions	20
Table 4.1 Accuracy values obtained using the maximizing the difference between sensitivity and 1 – specificity (Max TPR & TNR) and Closest Distance to Top-Left threshold approaches for D ^b and K ^b models across varying peptide lengths (8mer to 11mer). Average accuracy values are also presented for comparison.	43

LIST OF FIGURES

Figure 3.1. Example of BLOSUM substitution matrix [37]	22
Figure 3.2. The array representing the 9mer peptide, AAIGNQLYV, using One-hot encoding.	26
Figure 3.3. Leaky ReLU activation function	26
Figure 3.4. Binary cross-entropy loss function.....	27
Figure 3.5. The workflow of PREDBL6 web application	33
Figure 4.1. Prediction performance of various models using our EL+BA data as training and testing data. A) Linear Regression, B) Decision Tree, C) Random Forests, D) SVM, and E) ANN.....	37
Figure 4.2. H2-D ^b and H2-K ^b binding motifs. A) H2-D ^b binding motifs generated using MHC ligands (left) and thermostability data (right). B) The H2-D ^b 9mer motif reported by SYFPEITHI. C) H2-K ^b binding motifs generated using MHC ligands (left) and thermostability data (right). D) The H2-K ^b 8mer reported by SYFPEITHI.	40
Figure 4.3 The performance of the eight EL models assessed using AUC values for predicting peptide binding to H2-D ^b and K ^b alleles across various lengths, ranging from 8mer to 11mer. Each bar represents the average AUC value for the respective peptide length and allele type.	42
Figure 4.4. AUC values of the eight Thermostability models for predicting of binding stability to D ^b and K ^b alleles of various lengths (8mer to 11mer). Each bar represents the average AUC value for the respective peptide length and allele type.	45

Figure 4.5. Comparison of AUC Values for NetMHCpan-4.1, EL Model, and EL+TS Model. (A) AUC values for H-2D ^b allele across peptide lengths 8-11. (B) AUC values for H-2K ^b allele across peptide lengths 8-10.	47
Figure 4.6. Home page of PREDBL6	51
Figure 4.7. Tool page of PREDBL6	52
Figure 4.8. Instruction page of PREDBL6	53
Figure 4.9. Reference page of PREDBL6	54

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
AUC	Area Under the Curve
BA	Binding Affinity
C57BL/6	C57 BLACK6 Mouse
EL	Eluted Ligands
ER	Endoplasmic Reticulum
HLA	Human Leukocyte Antigen
HMM	Hidden Markov Models
IC50	Inhibitory Concentration (the concentration of peptides required to inhibit binding by 50%)
IEDB	Immune Epitope Database and Analysis Resource
IP	Immunoprecipitation
LC-MS/MS	Liquid Chromatography-Tandem Mass Spectrometry
MHC	Major Histocompatibility Complex
OHE	One Hot Encoding
pMHC-I	Peptide-MHC Class I
ROC	Receiver Operating Characteristic
SVM	Support Vector Machines
TAP	Transporter Associated with Antigen Processing
TN	True Negative
TP	True Positive
VACV	Vaccinia Virus

1. INTRODUCTION

1.1. T-cells, MHC Molecules, and Epitope-based Vaccines

Our body's immune system can be divided into two main branches, the innate immune system and the adaptive immune system, determined by the speed and specificity of the reaction. The innate immune system represents the physical barrier, like the skin and mucous membranes, that is the first line of defense, and it provides an immediate and non-specific response to pathogens. Its existence in even the simplest animals suggests its importance in survival. On the other hand, the adaptive immune system provides a more specific and targeted response to pathogens, symbolizing the immune system of higher organisms. The system is comprised of T cells, B cells, and antibodies, together offering the ability to "adapt" or "remember" previous encounters with pathogens; leading to enhanced response upon subsequent exposures.

The human adaptive immune response, characterized by specificity, involves specialized cells that detect non-self antigens and orchestrate targeted immune reactions [1]. Cytotoxic T cells, a vital component of the adaptive immune system, actively search for short antigenic peptides presented by major histocompatibility complex (MHC) class I molecules. The MHC class I antigen processing pathway involves several key steps: proteins are cleaved into shorter peptides by the proteasome, peptides are translocated into the endoplasmic reticulum (ER) via the transporter associated with antigen processing (TAP), further degradation of peptides may occur in the ER by aminopeptidases, peptides bind MHC molecules, and finally, peptide-MHC complexes are transported to the cell surface for recognition by CD8⁺ T cells [2]. MHC binding is

considered the most selective stage in T cell recognition. Peptides presented by MHC and recognized by T cells are referred to as T-cell epitopes. Note that not all MHC binding peptides can trigger T cell reactivity. Accurately identifying T cell epitopes speeds up the design and development of epitope-based vaccines [3,4].

1.2. Reverse Vaccinology

Computational approaches have been successful in designing epitope-based vaccines by precisely identifying vaccine targets. Deep learning has been proposed as an approach for effective vaccine design [6]. These approaches are top-down, and they do not resolve the host diversity and target mutability problem. We propose a bottom-up approach whereby deep learning is applicable to individual immunological profiles and the results provide individualized vaccine targets. The trained models then can be used to rapidly assess viral mutations and potential immune escape of viral variants. To explore this goal, we developed a model that targets a specific combination of MHC alleles and, for simplicity, deployed it on a mouse model.

Mouse models are extensively utilized in biomedical research to model human disease mechanisms due to several biological traits. Mouse models have brief gestation periods and short life spans. Their high fecundity and efficient breeding capabilities further enhance their suitability for scientific investigation. Another advantage of using mouse models is their similarity to humans in terms of anatomy, physiology, and genetics. In immunological research and vaccine development, mouse models are widely utilized due to their relatively less complex composition of the immunopeptidome [9].

Studies have shown that vaccines composed of synthetic peptides resembling cancer neoepitopes have resulted in efficient T-cell activities and killed cancer cells in both mouse models and human patients [10-13]. The C57 Black 6 (C57BL/6) mice are one of the earliest and most widely used inbred laboratory animals in biomedical research and vaccine development. C57BL/6 mouse expresses two MHC class I alleles, H2-D^b and -K^b [9].

Reverse immunology approaches involve the identification of immune targets through extensive bioinformatics screening of complete pathogenic genomes, followed by experimental validation [14]. Multiple online bioinformatics systems have been developed to predict peptides binding MHC alleles, including H2-D^b and H2-K^b alleles [3,15, 16]. Many of them were trained mainly based on MHC binding peptides identified using in vitro binding assays. MHC-peptide in vitro binding assays assess the binding of peptides to specific MHC molecules, a prerequisite for T-cell activation. With technological advancement, liquid chromatography-tandem mass spectrometry (LC-MS/MS) has been employed to physically detect naturally processed and presented peptides on the cell surface [17-19]. We refer to these experimentally determined, naturally processed peptides as eluted ligands. As more and more datasets of eluted ligands are becoming available, this gives us opportunities to build more accurate prediction models using more biologically relevant data while incorporating the antigen processing steps simultaneously [17-20].

Moreover, most existing tools only model a single step in the MHC class I antigen processing pathway, the binding between MHC molecules and peptides. While the

affinity of peptide-MHC-I (pMHC-I) complexes has traditionally been emphasized as a key determinant of immunogenicity, recent investigations suggest that stability may be a more accurate predictor [21]. The stability of pMHC-I complexes is essential for sustained presentation at the cell surface and induction of specific T-cell responses. The utilization of pMHC stability as a predictive marker for immunogenicity is substantiated by prior studies, including those focused on neoepitope prioritization. These investigations have consistently revealed a strong correlation between pMHC stability and peptide immunogenicity [22], with some studies suggesting that stability outperforms pMHC affinity in predicting immunogenicity.

We proposed to identify T-cell epitopes in C57BL mice by integrating contributing factors such as antigen processing and thermostability. We built a bioinformatics tool that predicts binding peptides of the MHC class I molecules H2-D^b and -K^b and its stability. Utilizing deep learning methods, we trained and rigorously validated the prediction models using naturally eluted MHC ligands. The prediction models are of high accuracy. This system is a prototype for exploring personalized targeting of vaccines for highly contagious and rapidly mutating pathogens. Because of the high combinatorial complexity of host-pathogen interaction of such viruses, deep learning represents a promising platform for improving personalized vaccine targeting.

1.3. Data Type (BA vs EL)

Two types of experimental data are commonly utilized to train prediction models for identifying MHC-restricted T cell epitopes. Binding affinity (BA) refers to the

strength with which a peptide molecule binds to a specific MHC molecule. BA data are often generated by in vitro MHC binding assays, and affinities are often reported as half-maximal inhibitory concentration (IC₅₀) values. The IC₅₀ value represents the concentration of peptides required to inhibit binding by 50%, with a lower value indicating stronger binding affinity. Prediction models trained using BA data can help identify peptides likely to be presented by MHC molecules. Eluted ligands (EL) are naturally processed peptides extracted from the MHC molecules after being bound to them, often using mass spectrometry techniques. These ligands contain information on the antigen processing steps that occur before MHC binding and offer direct insight into the peptides presented by MHC molecules during infections or antigen presentations.

In this study, we collected both EL and BA data from publicly available resources. EL assays can be conducted using various experimental techniques, making EL data more accessible. The majority of our training data were EL data. While BA data does not directly reflect in vivo presentation, it is useful in prioritizing peptides in training. Together, they encapsulate the interaction between peptides and MHC molecules. Many previous research studies that combine binding affinity data with eluted ligands yield more accurate models for T-cell epitope prediction. Thus, we incorporated both types of data to capture a more comprehensive picture of the peptide-MHC interactions.

1.4. Thesis Statement

This thesis project aims to develop an online computational system (PREDBL6) based on Artificial Neural Network (ANN) techniques to accurately predict ligands that stably bind the two MHC class I molecules in C57BL/6 mice, H2-D^b and H2-K^b. By integrating prediction models for MHC ligands and thermostability, the system is a valuable tool for immunologists. It aids in identifying peptides with a high potential of inducing immune responses, thus facilitating the rational design of epitope-based vaccines. Validation results show that PREDBL6 is sensitive and specific with good predictive ability. Compared with NetMHCpan-4.1, known for its accuracy, our integrated model showcases superior predictive capabilities. PREDBL6 enables rapid selections of potential T-cell epitopes. By narrowing the search space for immunologists, it reduces the need for extensive wet lab validation experiments, speeding up the rational design of epitope-based vaccines. With its user-friendly web interface, PREDBL6 is designed to be intuitive and requires minimal training to navigate effectively, ensuring accessibility across various research settings. PREDBL6 stands out as the first online prediction system to simulate MHC binding while considering other antigen processing steps and thermostability in C57BL/6 mouse, a widely used laboratory model organism. PREDBL6 is publicly available at <http://met-hilab.org:3001/tool>.

1.5. Contribution of the Thesis

The original contribution of this thesis project includes the development of an integrated online prediction system, PREDBL6, that accurately predicts immunogenic

peptides that stably bind H2b Class I molecules and stimulate immune responses in C57BL/6 mice. PREDBL6 integrates relevant contributing factors, such as the antigen processing steps before MHC binding and thermostability, with the MHC binding predictions. Firstly, we collected the most up-to-date EL and BA datasets to train deep learning models for MHC class I epitope prediction tailored specifically for C57BL/6 mice instead of solely using MHC binding peptides. Secondly, we built binding stability prediction models using proprietary thermostability data generated by our collaborators. While most existing systems primarily focus on predicting MHC binding affinities, our approach provides a more comprehensive assessment of peptide immunogenicity. Through the integration of thermostability prediction alongside MHC ligand prediction, PREDBL6 enables rapid and more accurate selections of potential T-cell epitopes. PREDBL6 has the potential to significantly advance epitope prediction methodologies and improve the efficiency and effectiveness of epitope-based vaccine design and immunotherapy development efforts.

1.6. Organization of the Thesis

In this thesis, we detail the development of PREDBL6, a web-based T cell epitope prediction system that simulates MHC binding and considers other antigen processing steps and thermostability in a model organism, C57BL/c mouse. Chapter 1 serves as an introductory gateway, providing an overview of the project, foundational biological and vaccinology concepts, and a clear thesis statement. Chapter 2 conducts an extensive literature review, tracing the historical evolution of epitope prediction models. Chapter 3

delves into the collection of training, testing, and validation datasets, the data encoding methods, the methodology employed for training our artificial neural network models, and the model validation process. In Chapter 4, our binding motifs are presented and compared with existing ones, the performance of the prediction models is summarized and compared with NetMHCpan-4.1, and the functionality of the web-based application PREDBL6 is introduced. Chapter 5 interprets our results, discusses the findings, and describes the prediction system's implementation and maintenance plan. Chapter 6 provides a summary of our work and contribution, as well as outlines future directions and potential enhancements based on lessons learned. Chapter 7 catalogs the papers cited throughout the thesis.

2. LITERATURE REVIEW

2.1. Peptide-based T cell Vaccines

Epitope-based vaccines represent an advancement in vaccine design, offering precise targeting based on the fundamental understanding of the immune system's recognition of epitopes. These vaccines elicit robust immune responses against pathogens while minimizing the risk of inducing unwanted immune reactions. This targeted approach enhances the efficacy and safety of the vaccines. Not all protein antigen sites are equally immunogenic in the context of T-cell responses [23]. Immunodominance is the result of some epitopes eliciting more immunological responses than others. This is particularly significant to consider in the case of peptide vaccinations, which typically target a limited number of critical epitopes. Peptide-based vaccinations maximize the potential for off-target effects while inducing strong and antigen-specific immune responses by carefully choosing peptide sequences that the immune system recognizes. Peptide vaccines also have better safety profiles than traditional vaccinations as they usually contain short peptide fragments free of adjuvants or live pathogens, lowering the risk of negative responses. Furthermore, peptide-based vaccinations can be produced quickly and cheaply, which allows for scalability and broad dissemination. These characteristics provide peptide-based vaccinations as a compelling platform for the advancement of customized and precision medicine techniques, with prospective uses in cancer, autoimmune disorders, and infectious illnesses.

As an example, peptide-based vaccines have played a significant role in addressing the current COVID-19 pandemic by facilitating the development of vaccines

[24]. With the urgent need for effective vaccines to reduce the spread of the virus, traditional vaccine development timelines have been compressed drastically. Typically, vaccine development takes between 10 and 15 years, involving sequential phases of testing to ensure safety and efficacy. However, the urgency of the COVID-19 situation has led to unprecedented scientific collaborations and financial investments, accelerating vaccine development processes. In the case of COVID-19, the development of vaccines has involved simultaneous testing of multiple candidates and the parallel progression of various phases of clinical trials while maintaining stringent safety standards. Epitope prediction serves as a crucial initial step in vaccine development, as epitopes stimulate immune responses from B-cells and T-cells. Computational methods, such as support vector machines (SVMs), motif-based systems, and neural networks, play a pivotal role in epitope prediction. These methods enable the design of successful vaccines by identifying peptide sequences that bind to MHC molecules and trigger immune responses. For instance, SVMs categorize data into binders and non-binders, while Hidden Markov Models (HMMs) help recognize peptide patterns with binder-like properties. By utilizing computational approaches, researchers can expedite the identification of potential vaccine candidates and accelerate the vaccine development process. Additionally, the selection of antigens based on predicted epitopes is crucial for vaccine design, ensuring the development of vaccines that effectively target the virus and induce robust immune responses. In this context, *in silico* methods offer a rapid and efficient means of identifying promising vaccine candidates, thereby contributing to the timely development of vaccines to combat infectious diseases such as COVID-19.

2.2 Existing Computational Systems

BIMAS & SYFPEITHI

SYFPEITHI and BIMAS are two of the earliest bioinformatics tools used for predicting peptide binding to MHC molecules [25,26]. Both tools provide web-based user interfaces for users to run predictions. SYFPEITHI is also a comprehensive database containing MHC ligands, epitopes, and detailed annotations of the sequences. SYFPEITHI incorporated these experimental data to develop prediction models, considering factors such as peptide sequence motifs, anchor residues, and binding affinities to predict pMHC binding. BIMAS adopted a mathematical algorithm to forecast peptide binding to MHC molecules based solely on peptide sequence information and MHC allele preferences. These two systems have been continuously updated and are still in use today. However, most recent benchmark studies have shown that newer methods generally outperform BIMAS and SYFPEITHI significantly [16, 27].

PREDBALB/C

Among the pMHC-I binding prediction tools tailored for mouse models, PREDBALB/c stands out as the first online computational system for the prediction of peptides binding to a complete set of MHC molecules in the BALB/c mouse [28]. The H2d haplotype consists of three MHC class I molecules, H2-K^d, H2-L^d and H2-D^d, and two class II molecules, I-E^d and I-A^d. By focusing on a complete organism, REDBALB/c enables users to identify a complete set of predicted targets of T-cell immune responses. Most existing pMHC binding prediction models for mice including

PREDBALB/c did not adopt ANNs, primarily due to the limited availability of training data, particularly for the H2-D haplotype. Consequently, these models predominantly relied on other models such as matrix-based approaches for prediction.

NetMHC-4.0

In the landscape of peptide-MHC binding prediction tools, the last non pan-specific prediction model in the NetMHC family is NetMHC-4.0 [29]. However, it's worth noting that the publication date for NetMHC-4.0 dates back to 2015, indicating that the model may not fully capture the wealth of eluted ligand data generated since then. Over the years, significant advancements have been made in immunological research, resulting in the accumulation of a vast amount of experimental data on peptide-MHC interactions. This growing body of data encompasses a broader range of MHC alleles and peptide sequences, offering valuable insights into the complexities of MHC binding preferences and peptide immunogenicity. Therefore, while NetMHC-4.0 is a valuable tool for epitope prediction, its predictive capabilities may be limited by the absence of more recent experimental data. As such, there is a pressing need for updated models that leverage the latest eluted ligand datasets to improve prediction accuracy and coverage.

MHCflurry 2.0

In the benchmark study conducted by Zhao and Sher (2018) [30], various MHC-binding predictors were evaluated using epitopes from different human MHC antigens, also known as the Human Leukocyte Antigen (HLA). MHCflurry emerged as one of the

top performers for MHC class I prediction. MHCflurry 2.0 represents a significant improvement over its previous version, MHCflurry [31,32]. One key highlight is the integration of more sophisticated ANN architectures, resulting in improved prediction accuracy and efficiency. Additionally, MHCflurry 2.0 boasts a pan-allele approach, allowing for predictions across a wide range of MHC alleles. Unlike our model, MHCflurry 2.0 is not tailored specifically to C57BL/6 mice and lacks a user-friendly interface, operating primarily through Python scripts. MHCflurry 2.0 provides strong prediction capabilities, but its publication date four years ago suggests it may not include the latest training data and could potentially benefit from updates. Furthermore, MHCflurry 2.0 introduces a distinct training approach, utilizing binding affinity values converted to a logarithmic scale ranging between 0 and 1. In their methodology, EL data is transformed into binding affinity values, treating positive EL data as weak binders in the training process.

NetMHCpan-4.1

NetMHCpan-4.1 is recognized as a robust and versatile bioinformatics tool for predicting peptide binding to MHC molecules. In the benchmark study by Paul et al. (2020) [16], NetMHCpan outperformed other tools in predicting MHC class I 9mers using naturally processed and eluted data from vaccinia virus (VACV) infection in C57BL/6 mice, which express the H-2D^b and H-2K^b MHC molecules relevant to our study. Developed as an extension of the well-established NetMHC framework, NetMHCpan-4.1 offers enhanced predictive capabilities by employing pan-specific

algorithms, allowing for simultaneous prediction across a wide array of MHC alleles. This approach significantly expands the tool's utility, facilitating analyses across diverse human populations and various MHC class I alleles. While the pan-specific approach of NetMHCpan-4.1 broadens its applicability, it also presents certain limitations. One notable drawback is the potential for decreased prediction specificity compared to allele-specific methods. Pan-specific algorithms inherently aim to generalize peptide-MHC binding predictions across multiple alleles, which may result in compromises in prediction accuracy for individual alleles.

NetMHCstab

NetMHCstab emerges as a notable tool for predicting peptide binding to MHC class I molecules, with a specific focus on kinetic stability and HLA [33]. The majority of data analyzed by NetMHCstab comes from the SYFPEITHI and IEDB databases. By incorporating both sequence-based features and structural information, NetMHCstab offers enhanced predictive capabilities, providing insights into the likelihood of peptide-MHC complex formation and persistence. Training data for NetMHCstab comprises 5509 9mers spanning 10 HLA alleles, encompassing a diverse range of peptide sequences and MHC specificities. The stability data utilized in model training were obtained through the scintillation proximity assay, which measures the half-life of peptide-MHC complexes. To facilitate model training and interpretation, the measured half-life values were transformed to a normalized scale ranging from 0 to 1, employing the equation $s = 2^{(-2/Th)}$, where s represents stability and Th denotes the measured half-life.

2.3. IEDB Database

The Immune Epitope Database and Analysis Resource (IEDB) compiles manually curated information on experimentally discovered B Cell and T cell epitopes found on various species, MHC binding peptides, and accompanying experimental settings [34]. Its contents were gathered primarily from literature from 1952 to now. We assembled our training data sets by collecting MHC ligand elution assay data from IEDB and enriched it with information from peptide processing tools [20].

Established in 2004, the IEDB is an openly accessible repository that houses a vast compilation of experimentally measured immune epitopes and analysis tools that assist researchers in processing those data. It is an openly accessible platform with its data obtained primarily from literature. As of January 2024, the IEDB has curated over 18,000 references, featuring an extensive collection of over 1,600,000 epitopes and approximately 1,400,000 B cell, T cell, MHC binding, and MHC ligand elution assays (both positive and negative). Due to continuous updates with new literature and data submissions, the IEDB provides researchers involved in vaccine design with a comprehensive and more accurate resource than relying solely on MHC binding predictions.

3. MATERIALS AND METHODS

3.1. Data Collection

3.1.1 The Collection of Eluted Ligands and MHC Binding Peptides

The training data for our EL models primarily consists of EL data with a smaller amount of BA data for peptide prioritization. When performing searches in the IEDB, BA and EL data were gathered separately. The following search criteria were adopted for EL data, Epitope: Linear peptide, Assay Type: MHC Ligand Elution Assay, Outcome: Positive, and MHC Restriction: H2-D^b (or H2-K^b) protein complex. The search result contained 96,838 assay entries as of November 11, 2023, with 44,565 for H2-D^b and 52,273 for H2-K^b.

Because the data in the IEDB were aggregated from literature, errors in data collection or contaminations in the original studies could lower the data quality. We discarded ligands identified only by one positive assay as part of the quality control process. Ligands with two or more positive assay records were kept in the data sets and labeled as MHC binders. Our final dataset included 3,395 positive H2-D^b ligands and 3,961 positive H2-K^b ligands of length 8-11 (Table 3.1). Additional peptides were extracted from the NetMHCpan-4.1 training datasets [20]. After comparing the two data sets and removing duplicates, 885 H2-D^b ligands and 1,187 H2-K^b ligands from the NetMHCpan-4.1 dataset were added to the training dataset (Table 3.1). We also introduced non-binders from the NetMHCpan-4.1 datasets and randomly sampled natural peptides. This helps us build a diverse set of negative examples, which many studies have proven to increase model accuracy.

Length	H2-D ^b ligands			H2-K ^b ligands		
	IEDB	NetMHCpan	Sum	IEDB	NetMHCpan	Sum
8	210	14	224	2411	646	3057
9	2357	585	2942	1292	395	1687
10	481	151	632	175	68	243
11	347	135	482	83	78	161
Sum	3395	885	4280	3961	1187	5148

Table 3.1 Eluted H2-D^b and H2-K^b ligands collected from IEDB and NetMHCpan-4.1 dataset.

The reason peptides with lengths between 8-11 are favored in many *in silico* MHC class I prediction tools has to do with the binding groove structures of MHC molecules. The MHC class I molecules typically have closed binding grooves, allowing 8-11mer peptides, while MHC class II molecules have open-ended binding grooves, allowing longer peptides with lengths between 13-25 [35]. Specifically, H2-K^b molecules prefer 8mer and 9mer peptides and H2-D^b molecules prefer 9mer peptides. Considering the availability of peptides with various lengths in the database, we focused on 8-11mer peptides when building computational models.

Additionally, we gather BA datasets from IEDB using the same query techniques (Table 3.2). In total, we added more than 10,000 peptides from BA experiments. We organized the data collected into three categories. Peptides with IC₅₀ values less than or equal to 50nm are considered strong binders. Those with IC₅₀ between 500nm – 50nm are considered weak binders. Peptides with IC₅₀ values more than 500nm are considered non-binders. Although the numbers of peptides in the BA and EL datasets are comparable, as shown in Table 3.2, most BA data are non-binders and weak binders. Hence, the binding peptides are dominated by EL data.

Length	H2-D ^b BA			H2-K ^b BA		
	Non-Binder	Weak Binder	Strong Binder	Non-Binder	Weak Binder	Strong Binder
8mer	1182	44	1	1102	649	578
9mer	1593	641	365	1507	439	186
10mer	825	104	29	486	50	9
11mer	223	50	4	158	28	6
Sum	5061			5198		

Table 3.2 Additional BA data for H2-D^b and H2-K^b alleles collected from IEDB.

3.1.2 Thermostability Data from Dana Farber Cancer Institute

In epitope-based vaccine design, we aim to predict immunogenic T cell epitopes i.e., peptides recognized by T cells and stimulate immune responses in the host. The models trained using BA and EL data are useful for predicting MHC binding peptides and MCH ligands. These predicted peptides and ligands are not necessarily immunogenic T-cell epitopes because other relevant biochemical factors are in play, such as thermostability. The thermostability assessments of MHC peptide binding evaluates the strength and duration of the interaction between the peptide and the MHC molecule under varying temperatures. The thermostability is relevant because the stability of the peptide-MHC complexes affects the efficiency of antigen presentation and the downstream activation of T cells [22]. Higher thermostability of peptide-MHC complexes allows longer interactions with T cell receptors, thus a higher likelihood of T cell activation.

Thermostability datasets were generated from the temperature gradient experiments done by our collaborators at the Dana Farber Cancer Institute. The experiments were specifically designed to investigate the stability of peptide-MHC (pMHC) complexes for H2-D^b and H2-K^b alleles under three temperature conditions

(37°C, 50°C, and 70°C) using the Mass Spectrometry technique. The binding peptides were isolated using immunoprecipitation (IP) techniques. Peptides with lower binding affinities tended to dissociate from the MHC molecules as the temperature increased, indicating their reduced binding stability on the MHC surface. The data enable us to perform a comprehensive thermostability analysis of MHC binding.

Table 3.3 summarizes H2-D^b and H2-K^b 8-11mer peptides categorized into three stability levels: low, medium, and high. Peptides identified exclusively at 37°C are considered to have the lowest stability, those identified exclusively at 37°C and 50°C are deemed to have medium stability, and those identified at 70°C only are deemed to have high stability. We ended up with over 3,000 H2-D^b binding peptides and over 5,000 H2-K^b binding peptides. These data were used to train the thermostability predictor model. Incorporating the stability dynamics of pMHC complexes into our prediction system enables the identification of peptides most likely to induce immune responses.

	H2-D ^b			H2-K ^b		
Length	Low Stability	Med Stability	High Stability	Low Stability	Med Stability	High Stability
8mer	145	86	12	2950	545	299
9mer	911	928	74	1128	137	85
10mer	284	240	35	198	40	53
11mer	223	166	43	100	21	39
Sum	1563	1420	164	4376	743	476
Total	3147			5595		
	37°C Training Label: 0.3	50°C Training Label: 0.6	70°C Training Label: 1.0	37°C Training Label: 0.3	50°C Training Label: 0.6	70°C Training Label: 1.0

Table 3.3 Peptide counts of the H2-D^b and H2-K^b stability dataset across various lengths and temperature conditions

3.2 Data Transformation

3.2.1 One-Hot-Encoding

When building machine learning models, one of the most important features is the amino acids in peptides, which are categorical data and cannot be directly used in models that expect numerical input. We tried various encoding methodologies to represent peptides in a format recognizable by machine learning methods. One widely used encoding method is One Hot Encoding (OHE).

OHE transforms categorical data, such as amino acids within peptide sequences, into binary arrays of zeros and ones that can be processed by machine learning models in a non-biased way. The principle behind one-hot encoding involves representing each unique category, in this case, individual amino acids, as a binary vector. In this vector, only one element is "hot" (assigned the value 1), while all other elements are "cold"

(assigned the value 0).

As there are 20 naturally occurring amino acids, we created a vector of length 20 for each peptide position to encode a peptide. Through OHE, each amino acid within the sequence is mapped to a unique binary vector of length 20. This ensures each category is represented independently of the others. It is a commonly used encoding method in machine learning.

3.2.2 BLOSUM Encoding

BLOSUM (BLOcks SUBstitution Matrix) encoding is another method widely adopted for representing amino acids in protein analysis and data transformation in machine learning (Figure 3.1). BLOSUM matrices capture biological information by specifying the similarity of one amino acid to another by a score [36]. The similarity score was generated from a database of trusted alignment. Each entry in a BLOSUM matrix represents the log-odds ratio of observing a particular amino acid substitution compared to what would be expected by chance. Positive values indicate amino acid substitutions that are more common than expected, suggesting functional similarity, while negative values indicate fewer common substitutions.

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

Figure 3.1. Example of BLOSUM substitution matrix [37]

3.3 Machine Learning Models

Linear Regression

Linear regression is a fundamental statistical method used to model the relationship between dependent and independent variables [38]. It assumes a linear relationship between the variables, where the dependent variable is a linear combination of the independent variables. Linear regression aims to find the best-fitting line (or hyperplane) that minimizes the sum of squared differences between the observed and predicted values.

Decision Tree

Decision Trees are non-parametric supervised learning methods used for classification and regression tasks [38]. They work by partitioning the feature space into

regions based on attribute values and constructing a tree-like model of decisions.

Decision Trees are versatile ML algorithms capable of fitting complex datasets and performing multioutput tasks. They also serve as the fundamental component of the Random Forest model.

Random Forest

We often achieve better predictions by aggregating the results of a group of predictors instead of relying solely on the best individual predictor. This learning method is known as the Ensemble method, where the group of predictors forms an ensemble [38]. Random Forest is an ensemble learning method that operates by constructing a multitude of decision trees during training and outputting the mode of the classes (classification) or the mean prediction (regression) of the individual trees. It offers high accuracy, handles large datasets with high dimensionality well, and is resistant to overfitting.

Support Vector Machine

Support Vector Machine (SVM) is a powerful supervised learning algorithm used for classification and regression tasks [38]. It works by finding the hyperplane that best separates the data points into different classes in the feature space. SVM aims to maximize the margin between classes, which helps in managing overfitting when combined with proper regularization and cross-validation, and it is capable of handling high-dimensional data effectively. SVM is particularly suited for small to medium-sized datasets and can handle linear and non-linear classification tasks.

Artificial Neural Networks

Artificial Neural Networks (ANNs) are computational models inspired by the biological neural networks of animal brains. They consist of interconnected nodes, called neurons, organized into layers. The neurons in one layer are connected to the neurons in the next layer, forming a network. Each connection between neurons has a weight associated with it, which determines the strength of the connection. ANNs are powerful machine learning models capable of learning complex patterns and relationships in data. They are particularly well-suited for tasks involving large amounts of data, such as image recognition, natural language processing, and predictive modeling.

One of the key advantages of ANNs is their ability to automatically learn features from raw data, eliminating the need for manual feature engineering. This makes them highly adaptable to different types of data and tasks. Additionally, ANNs are capable of learning non-linear relationships between input and output variables, allowing them to capture complex patterns that may not be apparent using traditional statistical methods.

We used Artificial Neural Networks to build prediction models due to their flexibility, scalability, and ability to handle high-dimensional data. ANNs have been employed in various bioinformatics tasks, and recent benchmark studies on MHC binding predictions have highlighted methods based on ANN architecture [16, 30]. We conducted further evaluations in Section 4.1 to compare the performance of ANN and the abovementioned machine learning models.

3.4 Study Design

Since we are using the eluted ligands that have been naturally processed and presented by MHC molecules on the surface of cells, it is straightforward that they are binders to MHC molecules once we confirm their existence. The epitope data we collected were identified using MS experiments; hence, they are all considered binders, and the peptides that were randomly generated are considered non-binders. These data were employed as training data in our eight EL models for H2-D^b and K^b alleles. Consequently, we can use the binding conditions as labels in our neural network model; we can assign 0 to non-binders and 1 to binders, which will become a binary classification problem.

The overall structure of our study consisted of the following steps that were employed in both our EL models and Thermostability models:

1. The datasets were stratified into training and testing sets, maintaining an approximate 80/20 ratio. This method ensures that the splitting process preserves the distribution of classes or characteristics within each subset, thereby ensuring that both the training and testing datasets accurately represent the overall dataset. The 80/20 ratio is widely adopted in machine learning to balance training and testing data. Additionally, we set the random seed to a constant value to ensure reproducibility.
2. We encoded each ligand in the training data into two NumPy arrays. The first array represents the amino acids in the ligand, as shown in Figure 3.2. After evaluating several amino acid encoding methods, we selected the one-hot encoding for simplicity and effectiveness [39]. The second array contains a binary label indicating

binding (1) or non-binding (0).

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

Figure 3.2. The array representing the 9mer peptide, AAIGNQLYV, using One-hot encoding.

3. We then constructed dense neural network models. The final model consists of one input layer and two hidden layers. The first hidden layer contains 128 neurons, and the second has 64 neurons with a Leaky ReLU activation function to introduce non-linearity and allow the learning of more complex patterns (Figure 3.3) [40].

$$\text{LeakyReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{if } x \leq 0 \end{cases}$$

Figure 3.3. Leaky ReLU activation function

Where α is a small positive constant, typically a small fraction. In our model, we set $\alpha = 0.01$ to allow a small gradient for negative values, allowing learning to occur even for negative inputs to avoid the vanishing and exploding gradients problem.

The output layer has a sigmoid activation function, guaranteeing that the output is between 0 and 1, making it useful for binary classification [41].

4. MHC ligands and binders tend to produce prediction scores close to 1 (binding), while non-binders tend to yield values close to 0 (non-binding). We compiled this model using the binary cross-entropy loss function [42] since both the sigmoid activation and binary cross-entropy loss functions are constructed for binary classification problems [38].

$$L = -[y * \log(p) + (1 - y) * \log(1 - p)]$$

Figure 3.4. Binary cross-entropy loss function

Where y represents the actual class label (either 0 or 1) of the binary classification problem, and p represents the predicted probability of the positive class. The loss function computes the logarithmic loss between p and y . When y is 1, the loss penalizes the model more if it predicts a low probability. When y is 0, the loss penalizes the model more for predicting a high probability for the positive class. The negative sign at the beginning of the equation converted it into a minimization problem. The goal is to minimize this loss function during the training to update the model's parameters and improve its predictive performance.

5. Batch normalization, a technique in deep learning, was applied to improve a neural network's performance. This is achieved by normalizing the mini-batch, then scaling and shifting the normalized values using learned parameters. Batch normalization

stabilizes the distribution of each layer's inputs during training, leading to faster and more stable convergence [43]. After trying various sizes for batch normalization, we chose batch size 32 as it produced the best performance.

6. We also adopted early stopping, a form of regularization to avoid overfitting [44]. The training process stops if the model's prediction performance on the validation set does not demonstrate improvement for a predefined number of epochs. We started training by running 100 epochs, and the model stopped in less than 20 epochs. After implementing the early stopping function with hyper-parameter patience being 10, we changed the epoch to 30. We trained the networks with 30 epochs and a batch size 32 and evaluated the model using the testing datasets.
7. We then fine-tuned the hyper-parameters to optimize the performance, including adjusting the number of hidden layers, the size of each layer, and the optimizer learning rate. We also tried various activation functions in hidden layers. We defined a callback function, such as Model Checkpoint, to record the best-performing model [38]. This way, we ensured that the best-performing model was recorded during training, which is helpful when trying out multiple sets of parameters and working with large datasets or when training takes a long time. We tried various optimizers and settled on Adam as it produced the best performance (Fig. 2) [45]. In stochastic gradient descent, the learning rate needs to be manually tuned. Adam computes individual learning rates for different parameters, resulting in adaptive learning rates.

3.5 Validation Dataset

To determine the predictive accuracy of our model on naturally processed peptides, we need a source that has comprehensively evaluated T-cell response in C57BL/6 mice. This external dataset should be relatively new and never exposed to our training models. Croft et al. investigated the immunogenicity of viral peptides presented by pMHC-I on the surface of infected cells [46]. They utilized peptide sequencing through high-resolution mass spectrometry to identify 172 pMHC-I derived from vaccinia virus-infected C57BL/6 mouse cells. The remaining peptides in the dataset, which were not detected by LC-MS/MS, included 46 VACV-derived H-2b-restricted peptides/epitopes from the Immune Epitope Database (IEDB), along with one entirely unpublished epitope and another mapped from a longer published sequence. Immune reactivity for each of the 221 peptides was tested eight times, which makes these epitopes a reliable source to test the predictive accuracy of pMHC models. Peptides that tested positive more than four times were classified as "major epitopes," while those that tested positive four or fewer times but tested positive at least one time were classified as "minor epitopes." Peptides that were never positive were classified as "nonimmunogenic." Among these peptides, 84 were classified as "major" epitopes, and 92 were classified as "minor" epitopes, with lengths ranging from 7 to 13 amino acids for the H2-D^b and K^b alleles.

To perform a more comprehensive evaluation of our model's performance, we looked into another benchmark study and generated all possible peptides from the VACV reference proteome. These peptides were considered non-immunogenic, hence, non-

binders, as they were not found in elution assays on infected cells. We also compared these randomly generated peptides with all the H2-D^b and H2-K^b positive binders in IEDB, and overlaps were removed to improve the quality of our validation dataset further. We can utilize this expanded dataset to assess our models' performance on peptides that were experimentally validated but not included in the training phase. We will apply this dataset to other current epitope prediction online tools to compare results. This should give us a comprehensive understanding of the performance of our model.

3.6 Evaluation Metrics

3.6.1 ROC & AUC

We adopted the Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) analysis as the evaluation metrics in this study. ROC curves are graphical representations used to assess the performance of binary classification models. They plot the true positive rate (sensitivity) against the false positive rate (1-specificity) at various threshold settings. This provides additional insight into the model's classification capacity. ROC curve allows for visualizing a model's ability to discriminate between true positives and false positives at different thresholds.

AUC quantifies the overall performance of a classification model by calculating the area under the ROC curve. AUC ranges from 0 to 1, where a “1” indicates perfect discrimination, suggesting the model can correctly classify all the instances. An AUC of 0.5 indicates the model is an equivalent of random guessing. AUC provides a single scalar value that summarizes the model's performance across all possible threshold

settings, making it a robust measure for comparing different models.

ROC curves and AUC were selected as evaluation metrics for several reasons. Firstly, they are widely used in machine learning and bioinformatics research, allowing for easy comparison with existing literature. Secondly, ROC curves provide a comprehensive visualization of a model's performance across various threshold settings, making them intuitive for interpretation. Additionally, AUC condenses the information from ROC curves into a single metric, facilitating quantitative comparison between different models. Overall, the combination of ROC curves and AUC analysis offers a robust and comprehensive approach for evaluating the predictive performance of classification models in this study.

3.6.2 K-fold Cross Validation

In addition to using ROC and AUC analysis, we employed the K-fold cross validations to evaluate the performance of the classification models further. K-fold cross-validation is a widely adopted machine learning technique for assessing models' generalization capability and reducing overfitting. K-fold cross-validation involves splitting the dataset into K equal-sized folds (five-folds in this instance), where K-1 folds are used for training the model, and the remaining fold is used for testing. This process is repeated K times, with each fold being used as the test set exactly once. The performance metrics are then averaged across all iterations to obtain a more reliable estimate of model performance.

There are several benefits to using K-fold cross-validation in this study. Firstly, it

provides a more robust estimate of model performance compared to a single train-test split. By training and testing the model on multiple subsets of the data, K-fold cross-validation reduces the variability in performance metrics. It provides a more accurate assessment of model generalization. Furthermore, K-fold cross-validation helps to ensure that the model's performance is not overly dependent on a particular training-testing split. This ensures a fairer evaluation as there is less chance the result is coming from an unbalanced training/testing dataset and chance.

Overall, the combination of ROC curves, AUC analysis, and K-fold cross-validation offers a comprehensive and rigorous approach to evaluating the predictive performance of classification models in this study.

3.7 Implementation

The web-based bioinformatics tool for epitope prediction was implemented using a combination of React.js and Python programming languages. React.js is a widely used JavaScript library for building user interfaces and application structures. It was chosen for its efficiency in creating interactive and dynamic web applications. This project utilized Python as the primary backend language for data processing, predictive modeling, and algorithm implementation. The choice of Python facilitated seamless communication between backend server operations and frontend web-based functionalities. Python packages NumPy and pandas were employed for efficient data manipulation, matplotlib and seaborn for data visualization, TensorFlow and Keras for building and training machine learning models, and the scikit-learn library (sklearn) for

machine learning utilities and evaluation metrics. The overall workflow of the web-based application is summarized in Figure 3.5.

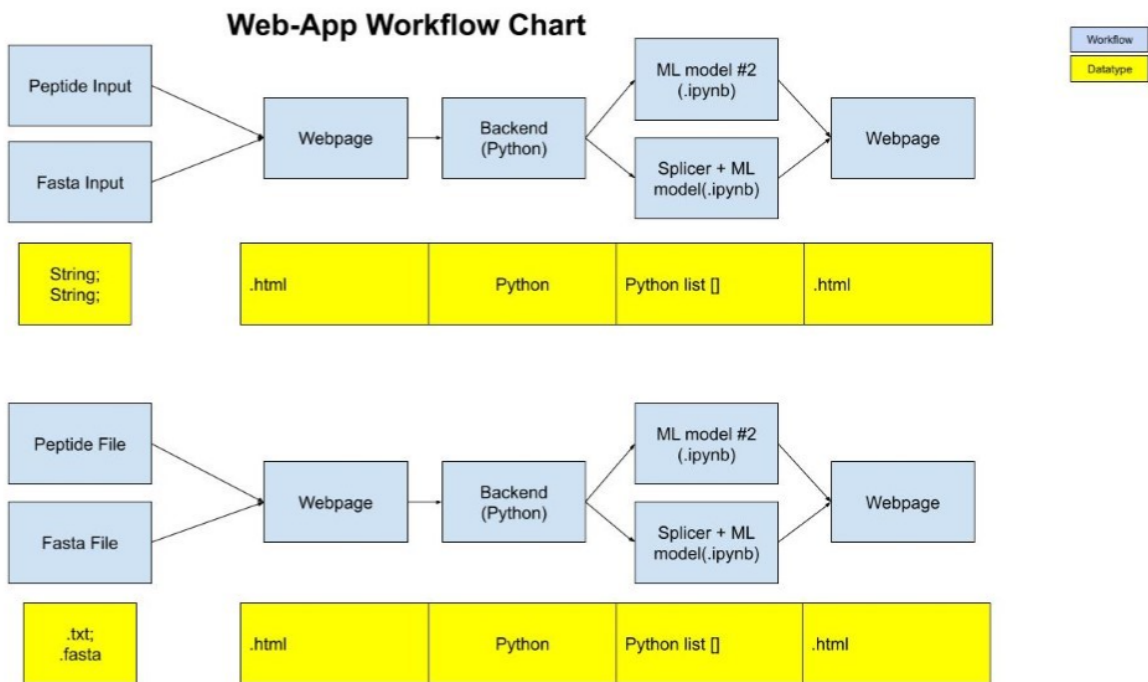


Figure 3.5. The workflow of PREDBL6 web application

4. RESULTS

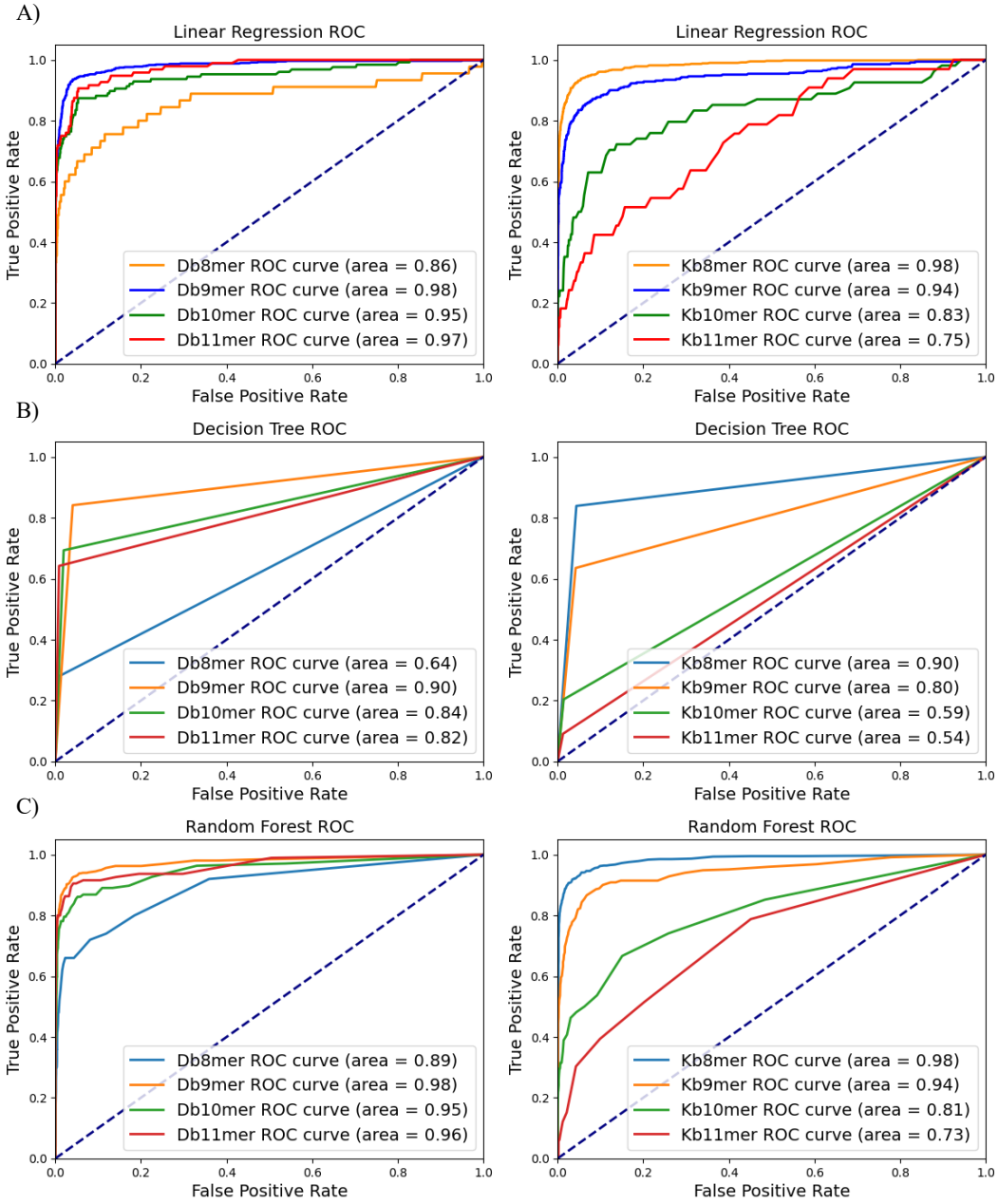
4.1 Comparative Analysis of Machine Learning Models

We compared various machine learning models to identify the most suitable approach for our epitope prediction task. Our evaluation encompassed a range of algorithms, including linear regression, Decision Trees, Random Forests, Support Vector Machines (SVM), and Artificial Neural Networks (ANN).

We assessed each model's performance and applicability to our epitope prediction task. Linear regression, a classic statistical method, provided a foundational benchmark by modeling the linear relationship between features and the target variable. Decision trees and random forests, with their intuitive tree-like structures, offer transparency and interpretability, making them ideal for understanding the importance of features in epitope prediction. Support Vector Machines (SVM) are powerful for classification tasks, especially in high-dimensional feature spaces, offering flexibility in capturing non-linear relationships. Lastly, Artificial Neural Networks (ANN) excel in capturing intricate patterns and nonlinear relationships, making them well-suited for epitope prediction where features may have complex interactions.

We used our EL+BA dataset, as described in Section 3.1, to conduct both training and testing tasks for these machine learning models. Subsequently, we employed the ROC & AUC evaluation metrics, as outlined in Section 3.6.1, to assess the performance of the different machine learning models on H2-D^b and -K^b data across peptide lengths 8-11 (Figure 4.1). As anticipated, ANN achieved the highest AUC score, averaging 0.91. While Linear Regression and Random Forests demonstrated similar performance (AUC =

0.9) compared to ANN, we chose ANN due to its capability to capture more intricate patterns and adapt to the nonlinear nature of epitope data, aligning closely with the complexity of our prediction needs.



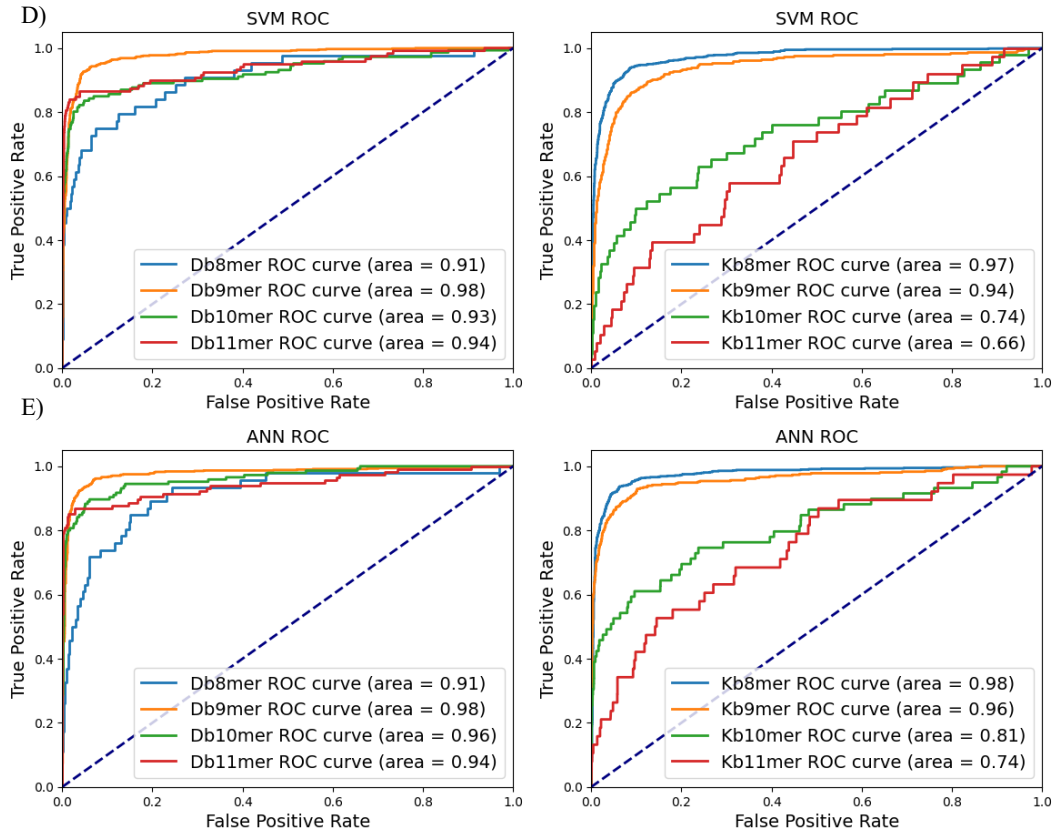


Figure 4.1. Prediction performance of various models using our EL+BA data as training and testing data. A) Linear Regression, B) Decision Tree, C) Random Forests, D) SVM, and E) ANN.

4.2 MHC Motifs

In addition to MHC binding prediction, SYFPEITHI is also one of the earliest online databases that capture information on MHC binding peptides, MHC binding motifs, and anchor positions. [47]. We used SYFPEITHI motifs as references and compared them with our ligand motifs. We first generated sequence logos using WebLogo3 [48] based on the EL and BA data described in Tables 3.1 and 3.2. These data make up the binding motifs for our EL models. We then generated sequence logos for our

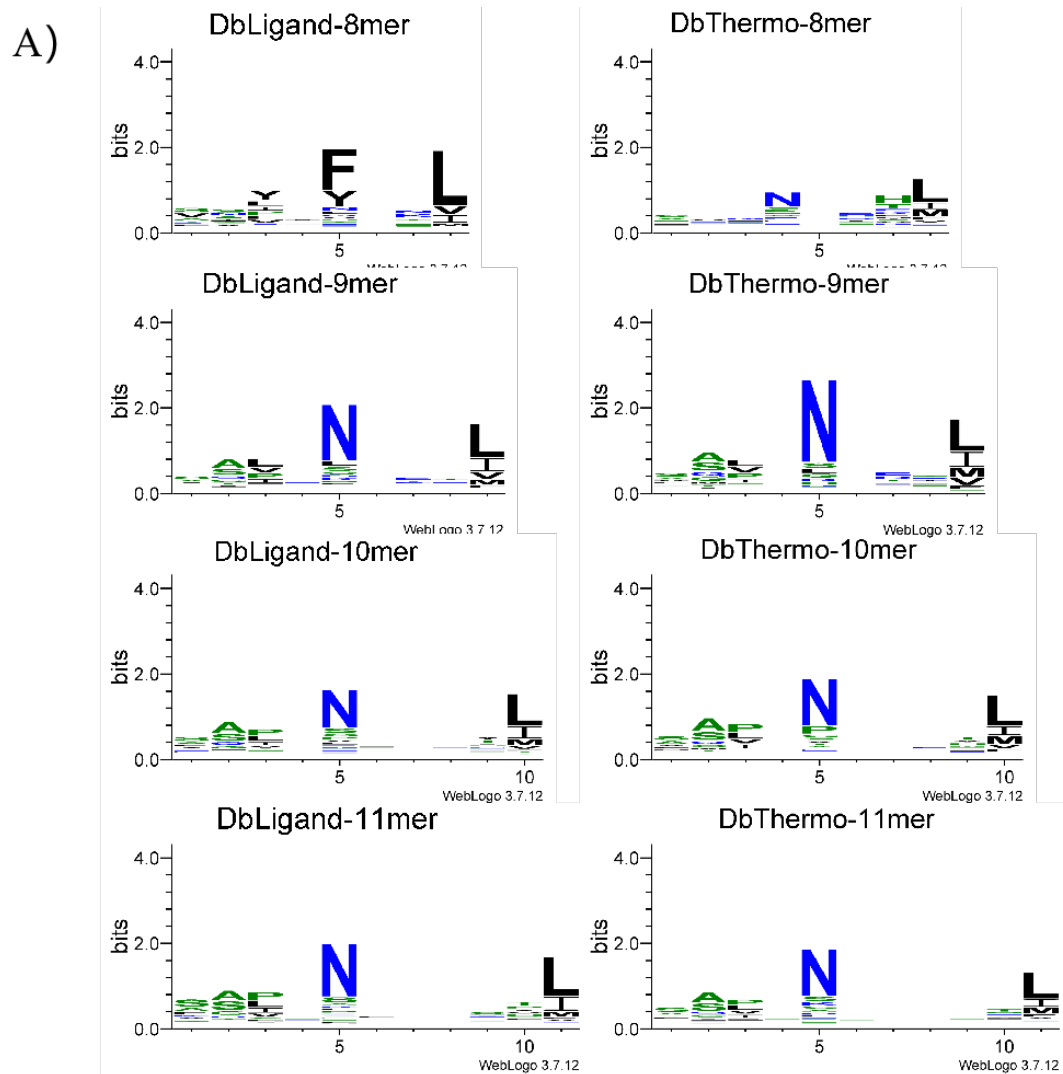
Thermostability models using data found in Table 3.3. These motifs were compared with binding motifs collected from SYFPEITHI (Figure 4.2).

As shown in Table 3.1, most of the H2-D^b ligands are 9mer peptides, while those for H2-K^b are primarily 8mer peptides. Figure 4.2 B) and D) are motifs for H2-D^b 9mer binders and H2-K^b 8mer binders reported by SYFPEITHI.

All three H2-D^b 9mer binding motifs, namely the SYFPEITHI motif, the ligand motif, and the thermostability motif, have positions 5 and 9 as primary anchor sites. Unlike the SYFPEITHI motif, the ligand and the thermostability motifs showed apparent selectivity at positions 2 and 3. The amino acid Asparagine (N) seemed more frequently observed at position 5 in the thermostability motif than the ligand motif. In our D^b 9mer, 10mer, and 11mer motifs, position 5 and the C-terminal position are primary anchor positions, while positions 2 and 3 consistently serve as auxiliary anchor positions. Our D^b 8mer binding motifs looked different from other D^b binding motifs. As shown in Tables 3.1 and 3.3, the counts of D^b 8mers were the lowest. Due to this limited 8mer data size, the reliability of the 8mer motifs is low.

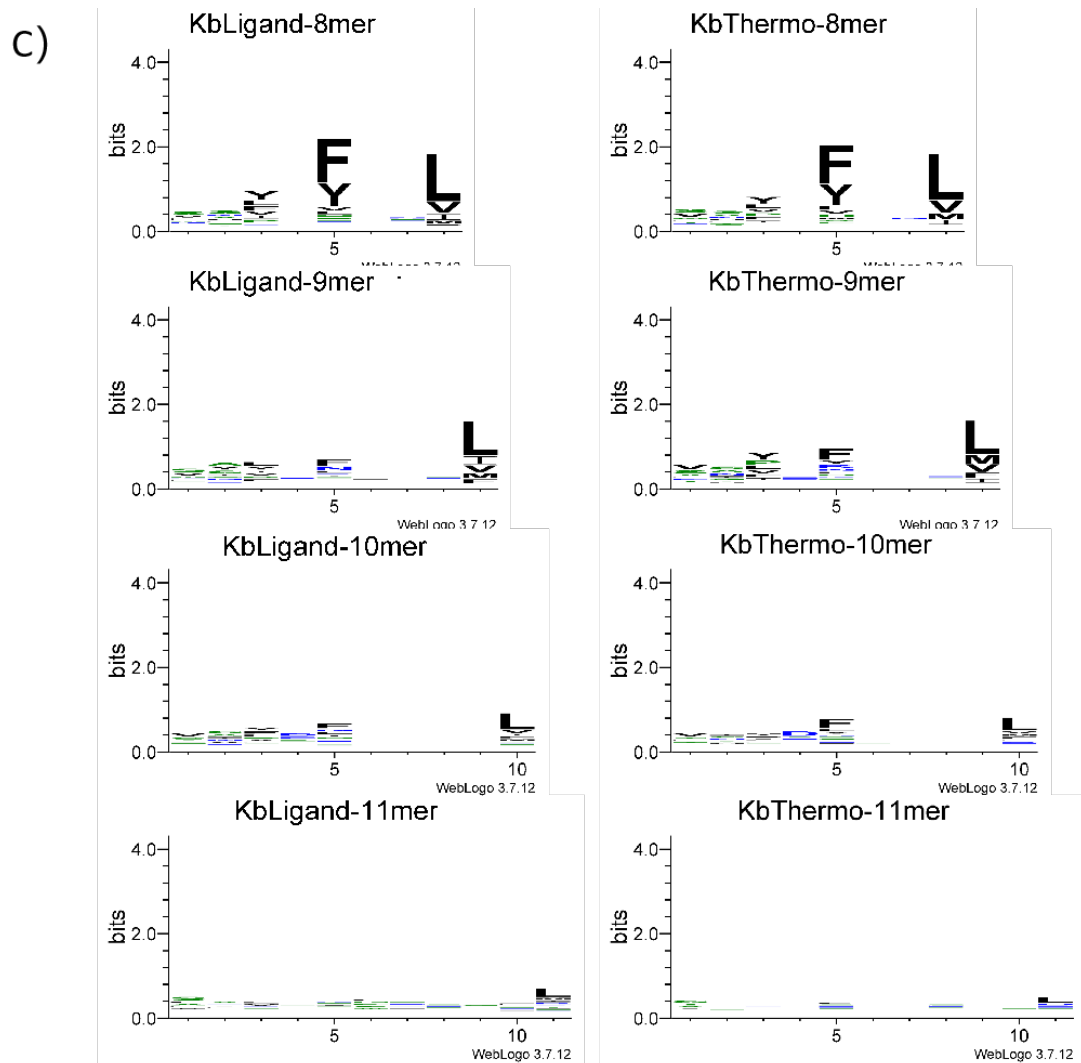
For H2-K^b 8mer binders, our motifs showed similarity to the SYFPEITHI one, with positions 5 and 8 being the primary anchor positions and position 3 as the auxiliary position. In our K^b 8mer, 9mer, and 10mer motifs, position 5 and the C-terminal position are primary anchor positions, while position 3 consistently serves as the auxiliary anchor position. For K^b binders, we noticed that as the peptide length increases, the significance of the motif pattern diminishes. This is especially evident for K^b 11mer motifs, where the sequence logo is relatively flat, and no clear anchor position exists. As shown in Tables

3.1 and 3.3, the counts of K^b 11mers were the least abundant. This scarcity of 11mer data underscored a challenge - the reliability of 11mer motifs was compromised due to their limited representation in the dataset.



B)

Db- Position								
1	2	3	4	5	6	7	8	9
anchors or auxiliary anchors								
N					M			
					I			
					L			



D)

Kb- Position								
1	2	3	4	5	6	7	8	9
anchors or auxiliary anchors								
Y	F	L						
	Y	M						
		I						
		V						

Figure 4.2. H2-D^b and H2-K^b binding motifs. A) H2-D^b binding motifs generated using MHC ligands (left) and thermostability data (right). B) The H2-D^b 9mer motif reported by SYFPEITHI. C) H2-K^b binding motifs generated using MHC ligands (left) and thermostability data (right). D) The H2-K^b 8mer reported by SYFPEITHI.

4.3 Internal Evaluation of EL Models

We built eight prediction models for H2-D^b and K^b ligands using 8mer to 11mer peptides. To assess the predictive performance of these models, we conducted five-fold cross-validation experiments for each model separately. The models were trained in an allele and length-specific manner, as described in section 3.4. The performance was evaluated in terms of the ROC curve and AUC values, as described in section 3.6. The average AUC values for the cross-validated models are shown in Figure 4.3.

The H2-D^b models showed excellent performance in predicting 9mer, 10mer, and 11mer binders, with AUC values of 0.97, 0.94, and 0.95, respectively. The model for 8mer peptides exhibited a lower AUC of 0.87, potentially attributed to the limited number of 8mer peptides in the training dataset. Additionally, the reduced presence of 8mer binders suggests that H2-D^b may have a lesser preference for 8mer peptides.

In contrast, the best performing H2-K^b model is the one for 8mer peptides, with an AUC value of 0.97. This is consistent with prior research on C57BL/6 MHC binding, which have reported a preference for shorter peptides in H2-K^b binding [49]. The models for 9mer, 10mer, and 11mer peptides also performed well, with AUC values of 0.93, 0.75, and 0.65, respectively. The lower AUC values for 10mer and 11mer peptides imply that these lengths are suboptimal for H2-K^b binding, potentially owing to the distinctive structural characteristics of the H2-K^b binding groove.

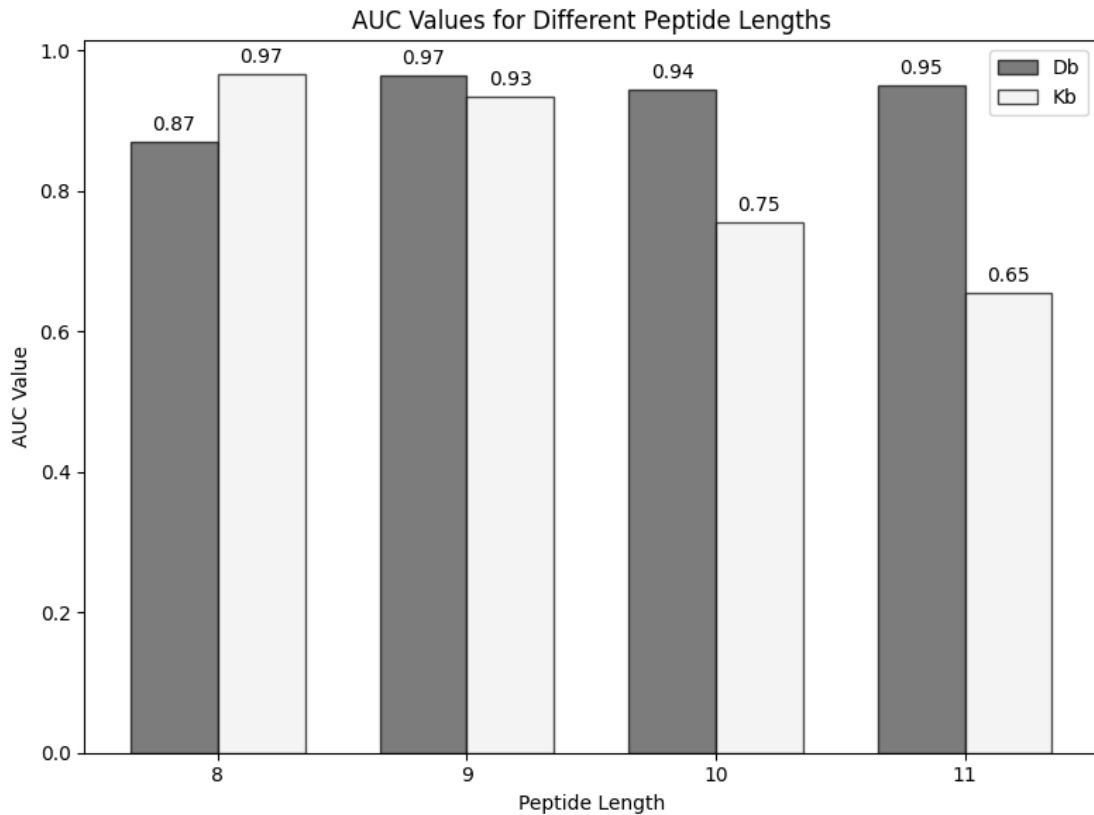


Figure 4.3 The performance of the eight EL models assessed using AUC values for predicting peptide binding to H2-D^b and K^b alleles across various lengths, ranging from 8mer to 11mer. Each bar represents the average AUC value for the respective peptide length and allele type.

4.4 Finding the Thresholds for EL models

When using the ligand prediction models, determining the optimal threshold is essential for classifying peptides as binders or non-binders to MHC molecules. The choice of thresholds impacts the models' ability to balance the true positive rate (sensitivity) and true negative rate (specificity), thus influencing the overall predictive performance.

Two approaches were adopted for the identification of thresholds. The first approach involves maximizing the difference between true positive rate (sensitivity) and

false positive rate ($1 - \text{specificity}$) across various threshold values along the ROC curve [50]. This approach aims to achieve the best balance between sensitivity and specificity, thereby optimizing the classification performance of our models. The second approach minimizes the Euclidean distance from each point on the ROC curve to the top-left corner, representing perfect classification. By selecting the threshold closest to the top-left corner, this approach can achieve a balanced prediction performance without favoring one metric over the other.

To determine the method that produces the optimal threshold, we rigorously evaluated each model's accuracy using thresholds generated by both methods. We found that the approach that maximized the difference between TPR and FPR yielded the best results. The average accuracy values are summarized in Table 4.1.

Length	Max TPR		Closest Distance to Top-Left	
	H2-D ^b ligands	H2-K ^b ligands	H2-D ^b ligands	H2-K ^b ligands
8mer	0.854	0.919	0.848	0.920
9mer	0.927	0.884	0.928	0.880
10mer	0.936	0.836	0.918	0.780
11mer	0.964	0.780	0.939	0.681
average	0.920	0.855	0.908	0.815

Table 4.1 Accuracy values obtained using the maximizing the difference between sensitivity and $1 - \text{specificity}$ (Max TPR & TNR) and Closest Distance to Top-Left threshold approaches for D^b and K^b models across varying peptide lengths (8mer to 11mer). Average accuracy values are also presented for comparison.

4.5 Internal Evaluation of Thermal Stability Models

To evaluate the overall performance of our Thermal Stability (TS) models, we first performed an internal evaluation of the eight TS models. We used 20% of the TS

dataset as the testing dataset and the remaining 80% as the training dataset. The AUC values are shown in Figure 4.4. This outcome is similar to that observed in the EL predictors. Among the models, the H2-D^b 9mer and H2-K^b 8mer exhibited the highest AUC values of 0.98 and 0.97, respectively. This underscores the robust binding capacity of p-MHC complexes at these specific peptide lengths.

Following closely, the H2-D^b 10mer and 11mer models exhibited AUC values of 0.94 and 0.93 respectively, while the H2-K^b 9mer model demonstrated an AUC of 0.91. These results highlight strong performance, suggesting their efficacy in distinguishing between stable and unstable peptide-MHC complexes. The H2-D^b 8mer model achieved an AUC of 0.87, indicating good, but not excellent, predictive capability for 8mer peptides in the H2-D^b context.

In contrast, the H2-K^b 10mer and 11mer models displayed subpar performance, with AUC values of 0.63 and 0.71, respectively. This is likely attributable to the limited training data available for these specific peptide lengths, potentially impeding the models' ability to learn effectively and generalize well. The lower performance in these models underscores the necessity for further optimization or the inclusion of additional data to improve their predictive accuracy.

Thermostability assessments of MHC peptide binding evaluates the strength and duration of the interaction between the peptide and the MHC molecule across different temperatures. Thermostability is relevant because the stability of the peptide-MHC complexes contributes to the efficiency of antigen presentation and the downstream activation of T cells.

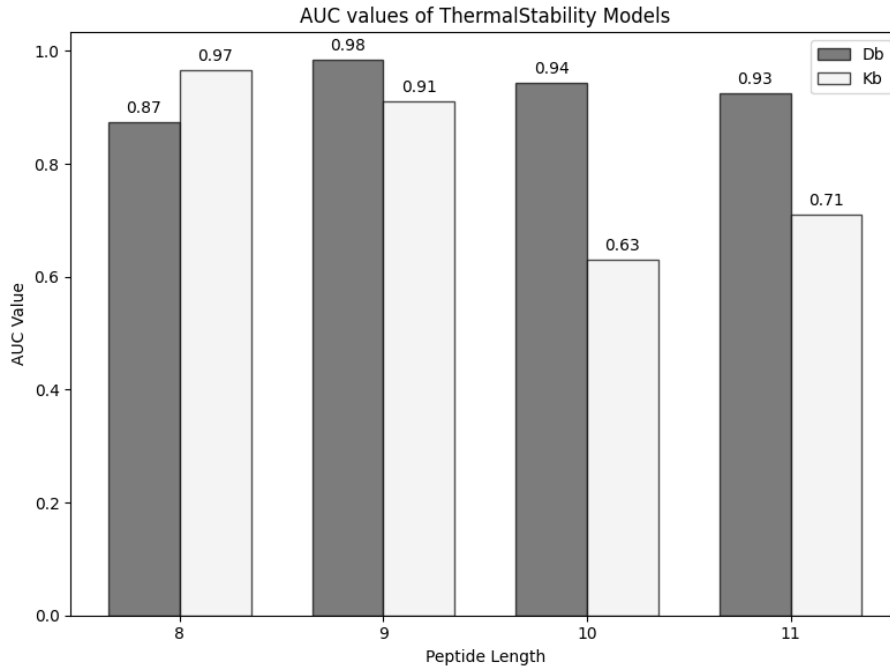


Figure 4.4. AUC values of the eight Thermostability models for predicting of binding stability to D^b and K^b alleles of various lengths (8mer to 11mer). Each bar represents the average AUC value for the respective peptide length and allele type.

4.6 External Validation

In this study, we employed an external validation dataset derived from the VACV to assess the predictive accuracy of our model on naturally processed peptides. We aim to compare the performance of our models with that of NetMHCpan-4.1, an online prediction tool validated by many benchmark studies to be one of the most accurate predictors. This dataset was used to validate our models' performance in predicting MHC ligands that were experimentally validated but not included in the training phase. NetMHCpan-4.1 is also the default recommended prediction method by IEDB. We input the validation dataset described in section 3.5 into the NetMHCpan-4.1 online tool and compared the prediction results with our EL models and the combination of our EL and

Thermostability models (EL+TS). The same AUC evaluation metrics were used in this evaluation.

The validation dataset was split by allele and peptide length. Additionally, we double checked to ensure that the validation dataset was not exposed to the training phase. To assess the predictive performance of each method, we evaluated the AUC performance in bar plots for each prediction approach across different peptide lengths (8–11 amino acids) and MHC class I alleles (H2-D^b, H2-K^b) (Figure 4.5). The AUC values were calculated individually for each peptide length and MHC allele, allowing for a comparative analysis of the predictive accuracy of each method. The average AUC values across the evaluated methods ranged from 0.58 to 0.97. Notably, our EL+TS model demonstrated the highest performance on this validation dataset, with an average AUC of 0.862. Meanwhile, the average AUC performance for the NetMHCpan4.1 model was 0.840. However, all three models exhibited poor performance in predicting peptides with longer peptide lengths, with this issue being particularly significant for the NetMHCpan4.1 model in predicting 10mer K^b ligands. Neither of our models performed as well as NetMHCpan4.1 on 11mer D^b ligands. Additionally, all three models demonstrated below-average performance on D^b 8mer ligands, with AUC values approximately at 0.70. Considering our training data availability, we only utilized 224 EL + 45 BA data for our EL models and 250 binders for our Thermostability model. The limited amount of training data could be a contributing factor to this undesired performance.

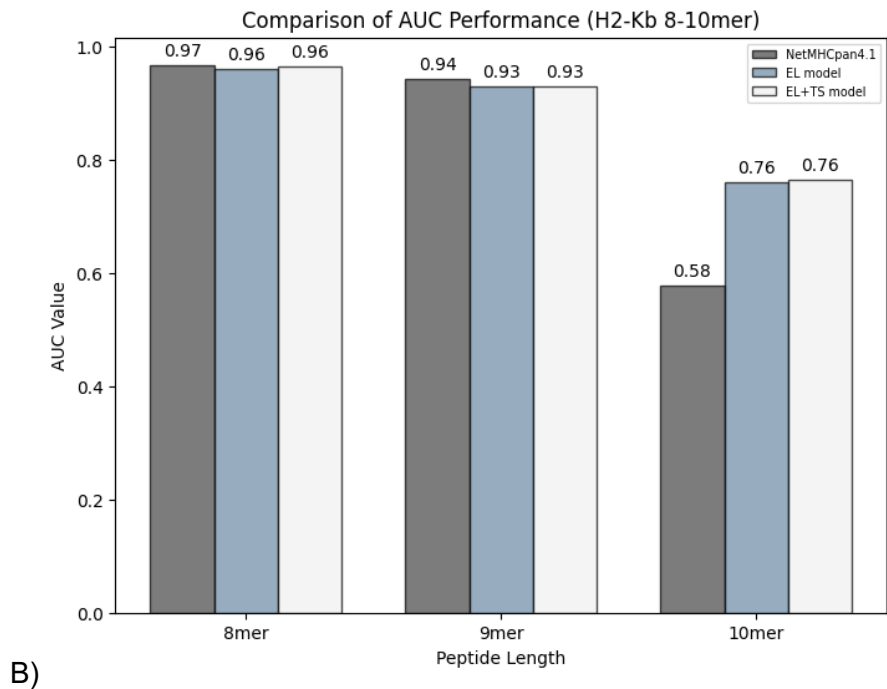
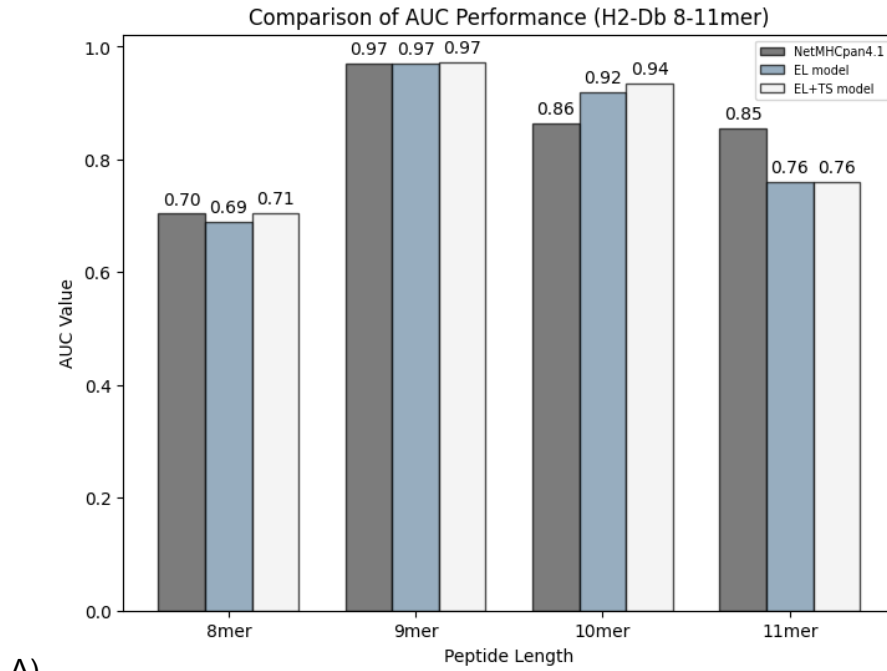


Figure 4.5. Comparison of AUC Values for NetMHCpan-4.1, EL Model, and EL+TS Model. (A) AUC values for H-2D^b allele across peptide lengths 8-11. (B) AUC values for H-2K^b allele across peptide lengths 8-10.

4.7 Implementation of PREDBL6

Integrating the eight EL models and eight stability models, we present the online web epitope prediction system PREDBL6 (Pred-black 6). The PREDBL6 online epitope prediction system was implemented using several web development technologies such as JavaScript, HTML, CSS, ReactJS, and Python. All the calculation and data processing tasks are processed on our in-house Linux server and the final results are sent back to the client. Using the web application, users have the freedom to select from the eight EL models and eight stability models for prediction.

4.7.1 Backend

We developed a Python Flask application for the web-based system's backend to effectively manage and process HTTP POST requests. Flask is a lightweight Python web framework widely recognized for its simplicity and flexibility, making it a popular choice for deploying machine learning models online. Utilizing Flask's compatibility with other Python libraries for machine learning and data processing, we integrated our ANN models into the Flask application. Flask receives and processes user input data, including uploaded files and other parameters. Subsequently, the Flask app forwards the file and parameters to the ANN models to perform predictions. Once the predicting process is completed, the backend retrieves the data, saves it in JSON format, and sends it to the frontend as a response for display purposes.

To reduce errors and improve usability, we developed a package for the machine learning code. We split the code into separate modules, including encoding and running ML, to enhance readability and promote code reusability. Additionally, we implemented

the Securefilename package to securely save the received file stream into a FASTA or peptide format for machine learning processing. Utilizing this library enhances the security of user uploads and ensures that they are normalized and sanitized for safe processing.

4.7.2 Frontend

The web-based system's frontend hosts our user-friendly web application developed using ReactJS and Tailwind CSS. ReactJS leverages Virtual DOM, enabling swift and efficient UI updates, while Tailwind CSS adopts a Utility-first methodology, providing a suite of predefined utility classes for styling elements. This approach streamlines development and maintenance, allowing for the creation of adaptable layouts and components suitable for diverse screen sizes and devices. Furthermore, we leveraged React components to enhance reusability, modularity, and organization within the web application. Components encapsulate their own logic, markup, and styles, facilitating unit testing and future maintenance. For instance, we created reusable components such as navigation bars, tables, and text boxes, enhancing flexibility and ease of maintenance. Additionally, we separated the view and main files, with UI-related code defined in the component's render method, while behavior, state management, and logic are handled separately. This separation promotes improved code organization and maintainability.

In our web application development, we used one of the built-in React hooks, useState, to create a user-friendly and responsive interface. We were able to store user-selected parameters and utilize event handling methods to manage changes to these parameters. For instance, we implemented an input-type state to dynamically adjust the

UI based on the user's selection between FASTA and Peptide formats. Additionally, we employed the React Router library to define application routes, simplifying navigation implementation and allowing for clear separation between tool pages and introductory content. Furthermore, we used FormData to handle complex data structures, such as file uploading, and utilized the Fetch API to make HTTP requests. By collecting user data, including uploaded files and parameters such as type, length, and allele choices, using FormData, we could efficiently manage data process and communication.

4.7.3 Implementation

When deploying the application content to the school server, we utilized Nginx and Docker. Nginx is known for its high performance and efficient handling of numerous concurrent connections. By using Nginx to serve the frontend, we reduced the server's workload, thus freeing up resources and enhancing overall performance. Additionally, Nginx offers automatic response compression, reducing data transfer and improving page load times, particularly important for users with slower internet connections. We then utilized Docker to deploy the application on the server, employing the Docker Compose tool to coordinate between the frontend and backend components. Both Docker and Docker Compose were installed on both the development platform and the server. Docker streamlined the deployment process by packaging the application and all its dependencies into a Docker container. This approach ensured a consistent and reproducible environment across different systems, eliminating the need to install all Python libraries on the server.

4.8 Webpage

PREDBL6 system has three pages. The home page serves as the gateway, providing users an overview of the tool and its functionalities. It gives a brief description of the tool's purpose and features, highlighting its utility in predicting ligand and stability to C57BL/6 mouse MHC Class I molecules (D^b/K^b). Additionally, there are links and buttons that lead users to the MET HI Lab websites (Figure 4.6).

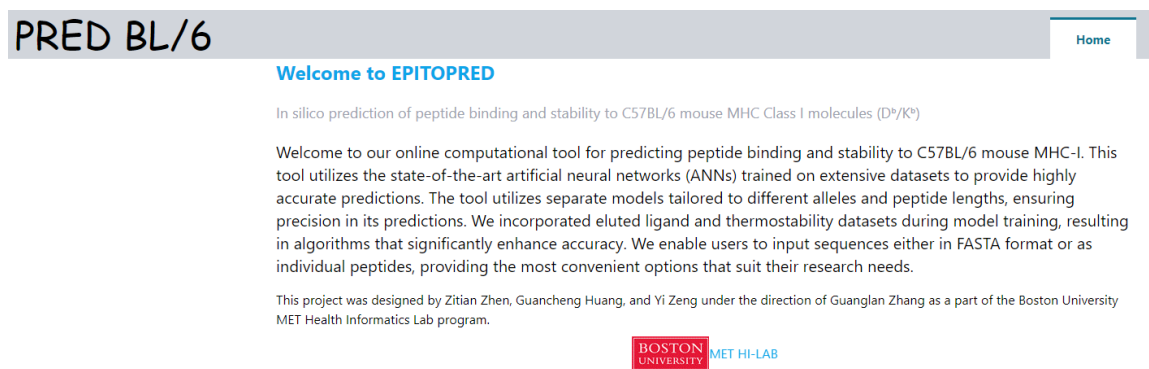


Figure 4.6. Home page of PREDBL6

The tool page is the main page where users can access the main functionality of the website, which is the ligand and stability prediction tool. Users can input peptide sequences either in FASTA format or individually, and the tool will provide predictions for ligand and stability to C57BL/6 mouse MHC Class I molecules. This page includes interactive elements for inputting sequences, selecting MHC alleles, and adjusting prediction parameters. Results are displayed in a user-friendly format, and binders are labeled in a separate column using the most optimal threshold. An overview of this main page is illustrated in Figure 4.7. When users want to incorporate our stability model in

prediction, they have this option by selecting the “use stability model” check box. Results will be displayed using scores combined with our EL and Thermostability model.

The screenshot shows the 'T cell prediction Tool' interface. At the top, there is a navigation bar with 'Home', 'Tool', and 'Reference' tabs. The main heading is 'T cell prediction Tool'. Below this, there is an 'INPUT TYPE:' dropdown menu set to 'peptide'. A text instruction reads: 'Paste a single or several peptides in PEPTIDE format into the field below:'. Below this is a large text input field with the placeholder 'Type here'. Underneath the input field, there is a link to 'upload a file in peptide format directly from your local disk' and an 'Upload file:' button labeled 'Choose File' with the text 'No file chosen'. The 'Parameters:' section includes a 'Length:' dropdown set to '8' and an 'Allele:' dropdown set to 'H2-Db'. There is a checkbox labeled 'use stability model' which is currently unchecked. On the right side, there are two buttons: 'Run' (blue) and 'clear' (green). At the bottom, a dashed box contains the text 'Prediction results will be displayed here when ready'. The Boston University MET HI-LAB logo is visible at the bottom center.

Figure 4.7. Tool page of PREDBL6

The instruction page in Figure 4.8 serves as a comprehensive guide for users, providing detailed information on all the buttons featured on the Tool page. To improve user experience, buttons are numbered on this page, allowing for clear and sequential guidance.

T cell prediction Tool

INPUT TYPE: **1. Choose Input Type**

Paste a single or several peptides in PEPTIDE format into the field below:

Type here **2a. Use textbox for FASTA or Peptide input**

... or **upload** a file in FASTA format directly from your local disk: **2b. Or upload FASTA or Peptide file**

Upload file: No file chosen

Parameters:

Length: 8 **3. Choose peptide length**

Allele: H2-Db **4. Choose allele**

use stability model **5. For inclusion of the Stability Model, click here**

6. Result will be displayed here




Figure 4.8. Instruction page of PREDBL6

The reference page is a resource hub containing information relevant to the tool's development and validation. It includes citations to relevant research papers, datasets, and methodologies used in training the prediction models. Users can access additional information about the underlying algorithms, model training procedures, and performance metrics. Currently, only the MET HI Lab hyperlink and our conference paper explaining the EL model are on the reference page. Additional department resources and future publications can be added to this page later (Figure 4.9).



Reference

Zhen, Z., Wang, Y., Keskin, D. B., Brusic, V., Chitkushiev, L., & Zhang, G. (2023). Deep Learning Models for Vaccinology: Predicting T-cell Epitopes in C57BL/6 Mice. In Zlateva, T., Tuparov, G. (eds) Computer Science and Education in Computer Science. CSECS 2023. (pp. 182–192). Springer. https://doi.org/10.1007/978-3-031-44668-9_14



Figure 4.9. Reference page of PREDBL6

5. DISCUSSION

5.1 EL+TS Models

Our C57BL/6 mouse T-cell epitope prediction system, PREDBL6, consists of two types of ANN models: the EL prediction models and the Thermostability models. The EL models were primarily trained using the EL dataset due to its extensive availability and direct reflection of the antigen presentation pathway. To augment the training data and improve the robustness of the models, we also incorporated a smaller amount of binding affinity data to cover the full spectrum of pMHC interaction.

The thermostability assessment of MHC peptide binding evaluates the strength and duration of the interaction between the peptide and the MHC molecule under varying temperatures. Studies have shown a positive correlation between thermostability and immunogenicity, as the stability of the peptide-MHC complexes affects the efficiency of antigen presentation and the downstream activation of T cells. Higher thermostability of peptide-MHC complexes allows for more extended interactions with T cell receptors, thus higher likelihood of T cell activation. Our collaborators at the Dana-Farber Cancer Institute performed temperature gradient experiments to investigate the stability of peptide-MHC complexes for H2-D^b and H2-K^b alleles under three temperature conditions, 37°C, 50°C, and 70°C, using the MS technique. The binding peptides were isolated using immunoprecipitation (IP) techniques. Peptides with lower binding stability tended to dissociate from the MHC molecules as the temperature increased, indicating reduced binding stability on the MHC surface. The data enables us to train the Thermostability models that will allow predicting the stability of peptides binding MHC

molecules. This enables a thorough examination of peptide-MHC interactions by altering the temperature variable, hence helping us to gain a better understanding of peptide stability.

Our analysis revealed a similar result between the binding motifs generated by both types of our models and those identified by SYFPEITHI. Further internal cross-validation unveiled that the EL model achieved an average AUC of 0.88, while the D^b allele models surpassed with an average AUC of 0.93, and the K^b allele models followed closely with an average AUC of 0.83. Similarly, the Thermostability model demonstrated a commendable average AUC of 0.87, with D^b allele models boasting an average AUC of 0.93, and K^b allele models recording an average AUC of 0.81. Notably, when focusing solely on peptide lengths 8 and 9, where the majority of immunogenic binders were identified, both models showcased outstanding average AUCs exceeding 0.93. The consistency of these results reinforces the robustness of our modeling approach. Particularly notable is the strong performance of the H2-D^b 9mer and H2-K^b 8mer models, highlighting specific peptide lengths that exhibit high stability and are likely to elicit a strong immune response. This aligns with established biological understanding and prior research, confirming our findings. Specifically, both D^b 9mer and K^b 8mer EL models achieved an AUC score of 0.97, and reached 0.98 and 0.97 for the Thermostability model, respectively.

The study conducted by Stevens et al. (1998) on peptide length preferences for rat and mouse MHC class I molecules demonstrated the influence of peptide length on stabilization. It suggested that H2-D^b has a preference for 9mer peptides, whereas H2-K^b

leans towards 8mer peptides [51]. Our findings and training results are consistent with the known MHC preferences for H2-D^b and K^b alleles reported in this study.

In comparison with the NetMHCpan-4.1 model, our model leverages newly released data and incorporates a larger training dataset. This allows our model to capture more diverse and comprehensive patterns in peptide-MHC binding interactions. Targeting the C57BL/6 mouse strain and specific peptide length enhances the relevance and applicability of our model for researchers working with laboratory mice. Another notable enhancement in our model is the incorporation of a Thermostability model. To further validate the effectiveness of our model, we tested on the VACV dataset that was never exposed to our models. We compared our models' performance with one of the most common epitope prediction tools, NetMHCpan-4.1. As illustrated in Figure 4.5, our model exhibited better overall performance on the VACV dataset. Specifically, the BA+TS model demonstrated superior predictive capabilities on the external validation dataset, surpassing the average AUC of the NetMHCpan-4.1 model. The average AUC value for NetMHCpan-4.1 was determined to be 0.85 for the D^b allele and 0.83 for the K^b allele. On the other hand, our best-performing BA+TS model achieved an average AUC of 0.85 for the D^b allele and 0.88 for the K^b allele. Utilizing our EL model alone resulted in an average AUC score of 0.84 for the D^b allele and 0.88 for the K^b allele. However, both NetMHCpan-4.1 and our models exhibited a decrease in performance with an increase in peptide length. The incorporation of the Thermostability model enhanced the performance of the predictions for all models, indicating the potential of integrating diverse data sources to improve predictive accuracy.

5.2 Webpage

In developing of our online tool PREDBL6, we integrated a comprehensive array of predictive models tailored to epitope binding and stability assessment. Our platform encompasses eight EL models and eight stability models, providing users with a versatile toolkit for epitope prediction. Users have the flexibility to select their desired MHC allele and can specify peptide lengths ranging from 8 to 11 amino acids. Additionally, our tool offers a choice between inputting sequences in FASTA format or as individual peptides, catering to the diverse needs of researchers and biologists. Users can incorporate the stability model into their predictions, enabling a more holistic assessment of epitope characteristics. By choosing to combine with the TS model, the prediction score is recalculated by adding 20% of the stability model. Binders are indicated in the output table using our default thresholds calculated for each allele and length. This enhances the interpretability of prediction results and provides a user-friendly interface.

We also provide additional links to our publications and department websites under the Resource Page for easy access. We aim to offer users a convenient method for further engagement and communication. Currently, our online tool is available at <http://met-hilab.org:3001/tool>, this tool will be fully migrated to the department's official server after thesis defense.

5.3 Online Tool Maintenance

We are determined to actively maintain the web-based system. We plan to periodically update the machine learning models through acquiring additional datasets

and retraining the models using them. The maintenance work also includes maintaining the Python source code used for data manipulation and prediction. The MET CS department maintains the web server, ensuring smooth functionality and addressing any technical issues that may arise. This collaborative effort ensures that PREDBL6 remains up-to-date and accessible to users, contributing to its long-term effectiveness and usability in the scientific community.

6. CONTRIBUTION AND FUTURE WORK

In this thesis project, we developed an integrated online system, PREDBL6, that accurately predicts ligands that stably bind the two MHC class I molecules in C57BL/6 mice, H2-D^b and H2-K^b. PREDBL6 distinguishes itself by incorporating the prediction models trained using the most up-to-date EL datasets and the Thermostability models explicitly tailored for C57BL/6 mice. While many prediction systems focus on MHC binding, our approach considers the stability of peptide-MHC interactions, providing a more comprehensive assessment of peptide immunogenicity. By integrating stability and ligand predictions, our system offers biology researchers a more accurate understanding of peptide-MHC interactions, enhancing the identification of immunogenic peptides.

While we have showcased the effectiveness and performance of PREDBL6 through external datasets and internal validation, it is essential to continuously update our training datasets and re-train the models accordingly to ensure their relevance and accuracy. With the exponential increase in the availability of epitope data and the advancements in experimental techniques, there is an ongoing need to leverage these resources to refine our models. Although our models performed well on the external dataset from VACV, we should always seek out additional external validation datasets to further validate the robustness of our model's predictions. For instance, Zhong et al. (2003) published a study containing a murine CTL epitope dataset of influenza viruses [52]. In their research, they conducted a genome-wide search using MHC class I binding algorithms and T-cell assays, identifying 16 epitopes recognized by CD8 T cells. These experimentally identified immunogenic D^b and K^b peptides, along with the results from

their stability experiments, can serve as a valuable validation source to assess the performance of our EL and Thermostability models.

To continuously improve the existing online prediction tool, we hope to include additional features and offer enhanced customization. One potential enhancement we are considering is the incorporation of threshold adjustment options, giving users the ability to personalize prediction in alignment with their specific research needs. This feature will enable users to select thresholds that prioritize either maximizing true positives (TP) or true negatives (TN), thereby enhancing the flexibility and precision based on users' objectives.

Another potential improvement in the web interface features involves allowing users to run predictions with multiple peptide lengths simultaneously. The current tool only allows users to select one peptide length at a time. By incorporating the functionality to select multiple peptide lengths concurrently, the tool stands to enhance user experience. This enhancement will empower users to expedite their analyses and streamline their workflow, maximizing productivity.

7. BIBLIOGRAPHY

1. Bonilla, F.A., Oettgen, H.C. Adaptive immunity. *Journal of Allergy and Clinical Immunology*, 125(2), S33–S40, (2010).
2. Shastri, N., Cardinaud, S., Schwab, S.R., Serwold, T., Kunisawa, J. All the peptides that fit: the beginning, the middle, and the end of the MHC class I antigen-processing pathway. *Immunological Reviews*, 207(1), 31–41, (2005).
3. Zhang, G.L., Keskin, B.D., Chitkushev, L. Extraction of Immune Epitope Information. In: Ranganathan, S., Nakai, K., Schönbach C. and Gribskov, M. (eds.), *Encyclopedia of Bioinformatics and Computational Biology*, vol. 3, pp. 39–46. Oxford: Elsevier, (2019).
4. Brusica, V., Petrovsky, N., Gendel, S.M., Millot, M., Gigonzac, O., Stelman, S.J. Computational tools for the study of allergens. *Allergy*, 58(11), 1083–1092, (2003).
5. Tregoning, J.S., Flight, K.E., Higham, S.L., Wang, Z., Pierce, B.F. Progress of the COVID-19 vaccine effort: viruses, vaccines and variants versus efficacy, effectiveness and escape. *Nature Reviews. Immunology*, 21(10), 626–636, (2021).
6. Abbasi, B.A., Saraf, D., Sharma, T., Sinha, R., Singh, S., Sood, S., Gupta, P., Gupta, A., Mishra, K., Kumari, P., Rawal, K. Identification of vaccine targets & design of vaccine against SARS-CoV-2 coronavirus using computational and deep learning-based approaches. *PeerJ*, 10, p.e13380, (2022).
7. Keshavarzi Arshadi, A., Webb, J., Salem, M., Cruz, E., Calad-Thomson, S., Ghadirian, N., Collins, J., Diez-Cecilia, E., Kelly, B., Goodarzi, H., Yuan, J.S. Artificial intelligence for COVID-19 drug discovery and vaccine development. *Frontiers in Artificial Intelligence*, 65, (2020).
8. Bagabir, S.A., Ibrahim, N.K., Bagabir, H.A., Ateeq, R.H. Covid-19 and Artificial Intelligence: Genome sequencing, drug development and vaccine discovery. *Journal of Infection and Public Health*, 15(2), 289–296, (2022).
9. Schuster, H., Shao, W., Weiss, T., Pedrioli, P.G., Roth, P., Weller, M., Campbell, D.S., Deutsch, E.W., Moritz, R.L., Planz, O., Rammensee, H.G. A tissue-based draft map of the murine MHC class I immunopeptidome. *Scientific Data*, 5(1), 1–11, (2018).
10. Banchereau, J., Palucka, K. Cancer vaccines on the move. *Nature Reviews. Clinical Oncology*, 15(1), 9–10, (2018).
11. Sahin, U., Türeci, Ö. Personalized vaccines for cancer immunotherapy. *Science*, 359(6382), 1355–1360, (2018).

12. Ott, P.A., Hu, Z., Keskin, D.B., Shukla, S.A., Sun, J., Bozym, D.J., Zhang, W., Luoma, A., Giobbie-Hurder, A., Peter, L., Chen, C. An immunogenic personal neoantigen vaccine for patients with melanoma. *Nature*, 547(7662), 217–221, (2017).
13. Keskin, D.B., Anandappa, A.J., Sun, J., Tirosh, I., Mathewson, N.D., Li, S., Oliveira, G., Giobbie-Hurder, A., Felt, K., Gjini, E., Shukla, S.A. Neoantigen vaccine generates intratumoral T cell responses in phase Ib glioblastoma trial. *Nature*, 565(7738), 234–239, (2019).
14. Rappuoli, R., Bottomley, M.J., D’Oro, U., Finco, O., De Gregorio, E. Reverse vaccinology 2.0: Human immunology instructs vaccine antigen design. *Journal of Experimental Medicine*, 213(4), 469–481, (2016).
15. Zhang, G.L., Sun, J., Chitkushev, L., Brusic, V. Big data analytics in immunology: a knowledge-based approach. *BioMed Research International*, 2014, 437987, (2014). <https://doi.org/10.1155/2014/437987>
16. Paul, S., Croft, N.P., Purcell, A.W., Tschärke, D.C., Sette, A., Nielsen, M., Peters, B. Benchmarking predictions of MHC class I restricted T cell epitopes in a comprehensively studied model system. *PLoS Computational Biology*, 16(5), e1007757, (2020).
17. Sarkizova, S., Klaeger, S., Le, P.M., Li, L.W., Oliveira, G., Keshishian, H., Hartigan, C.R., Zhang, W., Braun, D.A., Ligon, K.L., Bachireddy, P. A large peptidome dataset improves HLA class I epitope prediction across most of the human population. *Nature Biotechnology*, 38(2), 199–209, (2020).
18. Abelin, J.G., Keskin, D.B., Sarkizova, S., Hartigan, C.R., Zhang, W., Sidney, J., Stevens, J., Lane, W., Zhang, G.L., Eisenhaure, T.M., Clauser, K.R. Mass spectrometry profiling of HLA-associated peptidomes in mono-allelic cells enables more accurate epitope prediction. *Immunity*, 46(2), 315–326, (2017).
19. Truex, N.L., Holden, R.L., Wang, B.Y., Chen, P.G., Hanna, S., Hu, Z., Shetty, K., Olive, O., Neuberg, D., Hacohen, N., Keskin, D.B. Automated flow synthesis of tumor neoantigen peptides for personalized immunotherapy. *Scientific Reports*, 10(1), 723, (2020).
20. Reynisson, B., Alvarez, B., Paul, S., Peters, B., Nielsen, M. NetMHCpan-4.1 and NetMHCIIpan-4.0: improved predictions of MHC antigen presentation by concurrent motif deconvolution and integration of MS MHC eluted ligand data. *Nucleic Acids Research*, 48(W1), W449–W454, (2020).
21. Harndahl, M., Rasmussen, M., Røder, G., Pedersen, I. D., Sørensen, M. S., Nielsen, M., & Buus, S. Peptide-MHC class I stability is a better predictor than peptide

- affinity of CTL immunogenicity. *European Journal of Immunology*, 42(6), 1405–1416, (2012).
22. Jappe, E. C., Garde, C., Ramarathinam, S. H., Passantino, E., Illing, P. T., Mifsud, N. A., Trolle, T., Kringelum, J. V., Croft, N. P., & Purcell, A. W. Thermostability profiling of MHC-bound peptides: a new dimension in immunopeptidomics and aid for immunotherapy design. *Nature Communications*, 11(1), (2020).
 23. Malonis, R. J., Lai, J. R., & Vergnolle, O. Peptide-Based vaccines: current progress and future challenges. *Chemical Reviews*, 120(6), 3210–3229, (2019).
 24. Pollard, A.J., Bijker, E.M. A guide to vaccinology: from basic principles to new developments. *Nature Reviews. Immunology*, 21(2), 83–100, (2021).
 25. Rammensee, H., Bachmann, J., Emmerich, N., Bachor, O. A., & Stevanović, S. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics*, 50(3–4), 213–219, (1999).
 26. Parker KC, Bednarek MA, Coligan JE. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *Journal of Immunology*, 152(1), 163–175, (1994).
 27. Lundegaard, C., Lund, O., Buus, S., & Nielsen, M. Major histocompatibility complex class I binding predictions as a tool in epitope discovery. *Immunology*, 130(3), 309–318, (2010).
 28. Zhang, G., Srinivasan, K. N., Veeramani, A., August, J. T., & Brusica, V. PREDBALB/c: a system for the prediction of peptide binding to H2d molecules, a haplotype of the BALB/c mouse. *Nucleic Acids Research*, 33(Web Server), W180–W183, (2005).
 29. Andreatta, M., & Nielsen, M. Gapped sequence alignment using artificial neural networks: application to the MHC class I system. *Bioinformatics*, 32(4), 511–517, (2015).
 30. Zhao, W., & Sher, X. Systematically benchmarking peptide-MHC binding predictors: From synthetic to naturally processed epitopes. *PLoS Computational Biology*, 14(11), e1006457, (2018).
 31. O’Donnell, T. J., Rubinsteyn, A., Bonsack, M., Riemer, A. B., Laserson, U., & Hammerbacher, J. MHCFlurry: Open-Source Class I MHC Binding Affinity Prediction. *Cell Systems*, 7(1), 129–132.e4, (2018).

32. O'Donnell, T. J., Rubinsteyn, A., & Laserson, U. MHCFlurry 2.0: Improved Pan-Allele Prediction of MHC Class I-Presented peptides by incorporating antigen processing. *Cell Systems*, 11(1), 42–48.e7, (2020).
33. Jørgensen, K. W., Rasmussen, M., Buus, S., & Nielsen, M. NetMHCstab - predicting stability of peptide-MHC-I complexes; impacts for cytotoxic T lymphocyte epitope discovery. *Immunology*, 141(1), 18–26, (2014).
34. Vita, R., Mahajan, S., Overton, J.A., Dhanda, S.K., Martini, S., Cantrell, J.R., Wheeler, D.K., Sette, A., Peters, B. The immune epitope database (IEDB): 2018 update. *Nucleic Acids Research*, 47(D1), D339–D343, (2019).
35. Stevens, J., Wiesmüller, K., Walden, P., & Joly, E. Peptide length preferences for rat and mouse MHC class I molecules using random peptide libraries. *European Journal of Immunology*, 28(4), 1272–1279, (1998).
36. Henikoff, S., & Henikoff, S. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89(22), 10915–10919, (1992).
37. Mount, D. W. Using BLOSUM in sequence alignments. *Cold Spring Harbor Protocols*, 2008(6), pdb.top39, (2008).
38. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, (2019).
39. ElAbd, H., Bromberg, Y., Hoarfrost, A., Lenz, T., Franke, A. Wendorff, M. Amino acid encoding for deep learning applications. *BMC Bioinformatics*, 21, 1–14, (2020).
40. Maas, A.L., Hannun, A.Y., Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, (2013). https://ai.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf
41. Rumelhart, D.E., Hinton, G.E., Williams, R.J. Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536, (1986).
42. Topsøe, F. Bounds for entropy and divergence for distributions over a two-element set. *JIPAM. Journal of Inequalities in Pure & Applied Mathematics*. 2 (2). (2001)
43. Ioffe, S., Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Bach, F., and Blei, D. (eds.) Proceedings of the*

- 32nd International Conference on Machine Learning (ICML), 448–456, (2015).
<https://dl.acm.org/doi/10.5555/3045118.3045167>
44. Yao, Y., Rosasco, L., Caponnetto, A. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2), 289–315, (2007).
 45. Kingma, D.P., Ba, J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, (2014).
 46. Croft, N. P., Smith, A. M., Pickering, J., Sidney, J., Peters, B., Faridi, P., Witney, M. J., Sebastian, P., Flesch, I., Heading, S. L., Sette, A., La Gruta, N. L., Purcell, A. W., & Tschärke, D. C. Most viral peptides displayed by class I MHC on infected cells are immunogenic. *Proceedings of the National Academy of Sciences of the United States of America*, 116(8), 3112–3117, (2019).
 47. Schuler, M.M., Nastke, M.D., Stevanović, S. SYFPEITHI: database for searching and T-cell epitope prediction. *Methods in Molecular Biology*, 409, 75–93, (2007).
https://doi.org/10.1007/978-1-60327-118-9_5
 48. Crooks, G. E., Hon, G., Chandonia, J. M., Brenner, S. E. WebLogo: a sequence logo generator. *Genome Research*, 14(6), 1188–1190, (2004).
 49. Nielsen, M., & Andreatta, M. 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), (2016).
 50. Youden, W. J. Index for rating diagnostic tests. *Cancer*, 3(1), 32–35, (1950).
 51. Stevens, J., Wiesmüller, K., Walden, P., & Joly, E. Peptide length preferences for rat and mouse MHC class I molecules using random peptide libraries. *European Journal of Immunology*, 28(4), 1272–1279, (1998)
 52. Zhong, W., Reche, P. A., Lai, C., Reinhold, B. B., & Reinherz, E. L. Genome-wide characterization of a viral cytotoxic T lymphocyte Epitope repertoire. *The Journal of Biological Chemistry*, 278(46), 45135–45144, (2003).

CURRICULUM VITAE

