POEM: Prediction of immunogenic epitopes using mechanistic modelling of the MHC class I antigen processing pathway

Miles Weatherseed

St Anne's College

University of Oxford

A thesis presented in partial fulfilment of the requirements for

Doctor of Philosophy

Hilary 2024

Acknowledgements

Attributions: Eamonn Gaffney, Mark Coles, Tim Elliott and I contributed to the research of every chapter. David Arcia-Anaya carried out the experimental work required for Chapter 5.

Thanks are due to a great many people, without whose influence I am not sure I would have been able to complete this DPhil. First and foremost, to my academic supervisors, Eamonn Gaffney, Mark Coles and Tim Elliott, whose patience, support and expertise has helped me along this exciting journey from mathematician to computational biologist. To my research group, particularly Liam Brown, for their advice and friendship across the last few years. To the BBSRC for the financial support to carry out my research. To Grisha Szep and everyone at Synteny, for welcoming me and transforming my understanding of software development and machine learning.

To my incredible running coaches, Kyle Bennett, Mark Hookway and Matt Seddon, for tailoring my training and racing to fit around my academic goals, and under whose guidance I have been able to achieve more than I could ever have dreamt in the sport. To everyone at Vincent's Club, especially Gonzalo Hazard and Tom Smith, for financially supporting me and feeling like a family in Oxford. To all of my friends, for encouraging me when things got tough and for making me laugh every day.

Finally, to my parents, Simon and Marguerite Weatherseed, for being so proud and understanding of my desire to continue in academia, and to Alex Shipley, whose drive, intelligence and unfaltering compassion have inspired and nurtured me in equal measure throughout this journey.

Abstract

Existing machine learning algorithms predicting class I antigen presentation are fundamentally flawed due to the nature of the immunopeptidomics data used for their training. These models, rather than predicting antigen abundance, primarily indicate the presence of antigens on the cell surface. In this thesis, we integrate machine learning with mechanistic modelling to develop an enhanced model of the class I antigen processing pathway.

We begin by constructing a probabilistic model of epitope and precursor production by the proteasome, making use of existing algorithms to predict cleavage sites. We then develop a novel predictor of TAP binding affinity, *PanTAP*, which outperforms existing methods and forms accurate predictions across different mammalian species. Following this, we use a similar approach to train a predictive model of ERAP1 enzyme kinetics, enabling us to simulate the trimming of any potential substrate by ERAP1. These models are subsequently used to extend a previously validated systems biology model of peptide loading to MHC-I. This mechanistic model is validated using a study of SIINFEKL precursor processing and presentation in wild-type and ERAP1 knockdown cell lines, enabling us to infer the role of cytosolic aminopeptidases in epitope generation.

Finally, we use the validated mechanistic model to develop a new Predictor Of immunogenic Epitopes using Mechanistic modelling (POEM), employing a logistic regression trained on a dataset of neoantigens of known immunogenicity. POEM demonstrates superior efficacy on the training set and an independent dataset of GBM neoantigens. Furthermore, POEM accurately predicts the immunogenicity of pathogenic epitopes using a combined dataset from the IEDB, with its performance further validated through analysis of SARS-CoV-2 peptides across various HLA allotypes. Insights suggest that integrating source protein expression data could enhance POEM's predictions.

We conclude the thesis with a discussion of the results within the context of immunotherapy development and ideas for how our analysis may be further improved to provide clinical utility.

Contents

1 Introduction **1**

1.1 Research motivation 2

1.2 Biological review 4

1.2.1 The adaptive immune system 4

1.2.2 Class I antigen processing and direct presentation 7

1.2.2.1 Proteasomal cleavage 8

1.2.2.2 Cytosolic aminopeptidases 9

1.2.2.3 TAP translocation 9

1.2.2.4 ERAP1 and ERAP2 trimming 10

1.2.2.5 Peptide loading to MHC-I 12

1.2.3 CD8+ T-cell activation 13

1.3 Mathematical modelling review 14

1.3.1 Mathematical modelling in biology 14

1.3.2 Machine learning with peptides 17

1.3.3 Prediction of proteasomal cleavage 18

1.3.3.1 A history of proteasomal cleavage prediction . . . 18

1.3.3.2 Challenges to proteasomal cleavage prediction . . 19

1.3.4 Prediction of TAP translocation 20

1.3.4.1 History of TAP IC50 prediction 20

1.3.4.2 Limitations of existing TAP IC50 predictors . . . 20

1.3.5 Prediction of ERAP1 trimming 21

1.3.6 Prediction of MHC-I binding affinity 21

1.3.6.1 History of binding affinity prediction 21

1.3.6.2 Challenges to binding affinity prediction 23

1.3.7 Prediction of antigen presentation 23

1.3.7.1 History of antigen presentation prediction 23

1.3.7.2 Advantages of using immunopeptidomics to predict antigen presentation 24

1.3.7.3 Limitations of using immunopeptidomics to predict antigen presentation . . . 25
1.4 Prediction of CD8+ immunogenicity . . . 25
1.5 DPhil summary . . . 27

2 Prediction of proteasomal cleavage products 29
2.1 Introduction . . . 29
2.2 Methods . . . 31
2.2.1 Probabilistic model formulation . . . 31
2.2.1.1 Memoryless model . . . 32
2.2.1.2 Non-memoryless model . . . 32
2.2.1.3 Direction of protein entry . . . 35
2.2.1.4 Prediction of direction of entry . . . 36
2.2.2 Model fitting . . . 36
2.2.2.1 Parametrisation dataset . . . 36
2.2.2.2 Cleavage prediction algorithms . . . 37
2.2.2.3 Optimisation of model parameters . . . 38
2.3 Results . . . 40
2.3.1 Predictions of memoryless model . . . 40
2.3.2 Predictions of non-memoryless model . . . 40
2.3.3 Effect of substrate entry direction on product length distribution 41
2.3.4 Prediction of ovalbumin digestion . . . 41
2.4 Discussion . . . 45
2.4.1 Proteasomes inefficiently produce short peptides . . . 45
2.4.2 The direction of substrate entry affects specific product formation but not length distribution . . . 45
2.4.3 Choice of cleavage prediction algorithm . . . 46
2.4.4 Limitations . . . 47
2.4.5 Future work . . . 47
2.4.6 Concluding remarks . . . 47

3 Pan-species prediction of TAP binding affinity 49
3.1 Introduction . . . 49
3.2 Methods . . . 51
3.2.1 Construction of a training set . . . 51
3.2.1.1 Data sources . . . 51
3.2.1.2 Standardisation of units . . . 52
3.2.1.3 Data preprocessing . . . 52
3.2.2 TAP binding affinity predictive model training . . . 53

3.2.2.1 TAP pseudosequence . . . 53
3.2.2.2 Regression technique selection . . . 53
3.2.2.3 Peptide length standardisation . . . 54
3.2.2.4 Amino acid encoding . . . 54
3.2.2.5 Length encoding . . . 55
3.2.2.6 Hyperparameter tuning . . . 55
3.2.2.7 Cross-validation . . . 56
3.2.2.8 Ensemble construction and selection . . . 56
3.3 Results . . . 57
3.3.1 Effect of padding strategy . . . 57
3.3.2 Effect of amino acid encoding . . . 57
3.3.3 Results of ensemble construction . . . 57
3.3.4 Performance by species . . . 59
3.3.5 Performance by length . . . 59
3.3.6 DS613 benchmark against existing methods . . . 62
3.3.6.1 Models in the literature . . . 62
3.3.6.2 Human only model . . . 62
3.4 Discussion . . . 62
3.4.1 Optimal padding/trimming strategy aligns with anchor residues . . . 62
3.4.2 Pan-species prediction . . . 63
3.4.3 Ensembling of amino acid encodings . . . 63
3.4.4 Limitations . . . 64
3.4.5 Comparison to previous work . . . 64
3.4.6 Future directions . . . 64
3.4.7 Concluding remarks . . . 65

4 Prediction of epitope precursor N-terminus processing by ERAP1 66
4.1 Introduction . . . 66
4.2 Methods . . . 68
4.2.1 Construction of a training set . . . 68
4.2.1.1 Type of training data . . . 68
4.2.1.2 Data inclusion criteria . . . 69
4.2.2 Training . . . 70
4.2.2.1 Selection of regression techniques . . . 70
4.2.3 Amino acid encoding and padding/trimming . . . 70
4.2.3.1 Substrate length encoding . . . 71
4.2.3.2 Hyperparameter tuning . . . 71
4.2.3.3 Ensemble generation . . . 71

4.2.4 Conversion to Michaelis-Menten kinetics 72
4.2.4.1 Estimating Michaelis-Menten kinetic parameters . 72
4.2.4.2 Calibrating MM kinetics and trimming rates . . . 74
4.3 Results 74
4.3.1 Effect of padding strategy on predictive performance 74
4.3.2 Effect of encoding on predictive performance 75
4.3.3 Results of ensemble construction 75
4.3.4 Final model performance 77
4.3.4.1 Length bias 78
4.3.4.2 Benchmarking against ERAMER 79
4.3.5 Michaelis-Menten kinetics for ERAP1 79
4.3.6 Calibration curve for converting scores to catalytic rates . . 79
4.4 Discussion 81
4.4.1 Amino acid encodings are problem-specific 81
4.4.2 Limitations 82
4.4.2.1 Small training data set 82
4.4.2.2 Possible length bias 82
4.4.2.3 Application of *in vitro* to *in vivo* 83
4.4.2.4 ERAP1 polymorphism 83
4.4.3 Future work 83
4.4.4 Concluding remarks 84

5 Modelling of tapasin-assisted MHC-I loading 85
5.1 Introduction 85
5.2 Methods 86
5.2.1 The Dalchau model 86
5.2.1.1 Model overview 86
5.2.1.2 Extension of Dalchau model 87
5.2.2 Refitting the *Dalchau* model 92
5.2.2.1 Endogenous peptide 92
5.2.2.2 Conversion of concentrations 93
5.2.2.3 Assumptions of peptide-MHC abundance 93
5.2.2.4 Parametrisation dataset 95
5.2.2.5 Simulation of experimental data 96
5.2.2.6 Bayesian inference via MCMC 98
5.2.3 Tapasin dependence in the *Dalchau* model 100
5.2.4 Prediction of tapasin dependence 101
5.2.4.1 Measurements of tapasin dependence 101
5.2.4.2 MHC-I sequence processing 102

5.2.4.3 Amino acid encoding . . . 102
5.2.4.4 MHC-I structure prediction and encoding . . . 102
5.2.4.5 Regression training . . . 103
5.3 Results . . . 104
5.3.1 Extended *Dalchau* model inference . . . 104
5.3.2 Tapasin dependence in the extended *Dalchau* model . . . 108
5.3.3 Prediction of tapasin dependence . . . 110
5.3.3.1 Comparison of encodings and MHC-I representations 110
5.3.3.2 Correlation of tapasin dependence and prediction error . . . 111
5.3.3.3 Performance of best model . . . 111
5.4 Discussion . . . 113
5.4.1 Low SIINFEKL presentation in .220 cell line . . . 113
5.4.2 MHC-I on rate determines tapasin dependence in the extended *Dalchau* model . . . 115
5.4.3 Tapasin dependence cannot be accurately predicted by MHC-I sequence . . . 115
5.4.4 Single structure predictions do not improve model performance 116
5.4.5 Concluding remarks . . . 117

6 A mechanistic model of the antigen processing pathway 119
6.1 Introduction . . . 119
6.2 Methods . . . 121
6.2.1 Mechanistic model fitting . . . 121
6.2.1.1 Hearn data set . . . 121
6.2.1.2 Cytosolic aminopeptidases . . . 121
6.2.1.3 Off-rate measurement . . . 122
6.2.1.4 Parameterisation . . . 122
6.2.1.5 Model simulation . . . 123
6.2.1.6 Bayesian inference via MCMC . . . 124
6.2.2 Global sensitivity analysis . . . 125
6.3 Results . . . 127
6.3.1 Hearn fitting results . . . 127
6.3.2 Comparison with Schatz study . . . 133
6.3.2.1 Schatz parameter inference . . . 133
6.3.2.2 Comparison of Hearn and Schatz parameters . . . 133
6.3.3 Sensitivity analysis . . . 136
6.3.3.1 Antigen presentation is most sensitive to epitope parameters . . . 136

6.3.3.2 Aminopeptidases may play a destructive role in epitope generation . . . 136
6.3.3.3 Precursors significantly contribute to epitope presentation . . . 137
6.3.3.4 Longer precursors have little effect on presentation 137
6.4 Discussion . . . 138
6.4.1 Possible reasons for discrepancies . . . 138
6.4.1.1 Errors in parameter prediction . . . 138
6.4.1.2 Errors in parameter prediction . . . 139
6.4.1.3 Ubiquitin removal in cytosol . . . 140
6.4.2 Limitations of analysis . . . 140
6.4.2.1 Peptide diversity . . . 140
6.4.2.2 The role of ERAP2 . . . 142
6.4.2.3 Proteasomal predictions . . . 142
6.4.2.4 Lack of MHC-I diversity . . . 142
6.4.2.5 Semi-quantitative data . . . 143
6.4.3 Concluding remarks . . . 143

7 Prediction of immunogenic CD8+ epitopes through mechanism 145
7.1 Introduction . . . 145
7.2 Methods . . . 147
7.2.1 POEM input features . . . 147
7.2.1.1 MHC-I pseudosequence . . . 148
7.2.1.2 Peptide length encoding . . . 148
7.2.1.3 TCR contact sites . . . 148
7.2.1.4 Mechanistic prediction of antigen presentation . . . 150
7.2.2 POEM training . . . 151
7.2.2.1 PRIME-2.0 training set . . . 151
7.2.2.2 Data pre-processing . . . 152
7.2.2.3 Model training . . . 152
7.2.3 POEM testing . . . 153
7.2.3.1 GBM dataset . . . 153
7.2.3.2 BigMHC IEDB dataset . . . 154
7.2.3.3 SARS-CoV-2 dataset . . . 154
7.2.4 POEM benchmarking . . . 155
7.2.5 Performance evaluation . . . 157
7.3 Results . . . 158
7.3.1 POEM model development . . . 158
7.3.1.1 Comparison of proteasomal prediction algorithms 158

7.3.1.2 Comparison of MHC-I sequence representations . 159
7.3.1.3 Comparison of peptide sequence representations . 160
7.3.1.4 Logistic regression vs. multi-layer perceptron . . . 161
7.3.2 POEM performance 162
7.3.2.1 POEM improves prediction accuracy on PRIME-2.0 training set 162
7.3.2.2 POEM accurately identifies immunogenic GBM neoepitopes 164
7.3.2.3 POEM predicts pathogenic epitopes 164
7.3.2.4 POEM accurately identifies SARS-CoV-2 epitopes across a range of HLA types 165
7.3.2.5 Source protein expression can further enhance POEM predictions 166
7.3.3 Sources of POEM efficacy 167
7.3.3.1 Mechanistic model predictions 167
7.3.3.2 POEM ablations 169
7.3.3.3 Feature importance 170
7.4 Discussion 172
7.4.1 Efficacy of POEM 172
7.4.2 Limitations 172
7.4.2.1 TCR frequency 172
7.4.2.2 Binding affinity prediction 173
7.4.2.3 Rare HLA alleles 173
7.4.2.4 Computation time 173
7.4.3 Future directions 174
7.4.4 Concluding remarks 174

8 Discussion **176**
8.1 Research goals 176
8.2 Research summary 177
8.3 Limitations 179
8.3.1 Training data availability 179
8.3.2 Tapasin dependence 180
8.3.3 ERAP2 omission 180
8.3.4 Homogeneous validation datasets 181
8.3.5 POEM performance 181
8.4 Future work 182
8.4.1 Integration of protein expression data 182
8.4.2 Reducing bias in machine learning 183

8.4.3 Tumour evolution and immune escape 184
8.4.4 Future pandemics 184
8.5 Closing remarks 185

A Appendix **187**
A.1 Extended *Dalchau* model 187
A.2 Hearn et al. model 189
A.3 POEM mechanistic model 193
A.3.1 System of differential equations 193
A.3.2 Model parameters 196
A.4 POEM supplementary figures 197
A.4.1 MHC-I representations 197
A.4.2 Peptide representations 198
A.4.3 PRIME-2.0 dataset AUPR 199
A.4.4 SARS-CoV-2 dataset AUPR 200
A.4.5 POEM pMHC predictions comparison 201

References **202**

List of Abbreviations

ABC	ATP Binding Casette
ABM	Agent Based Models
ANN	Artificial Neural Network
APC	Antigen Presenting Cell
ATP	Adenosine Triphosphate
AUPR	Area Under PR Curve
AUROC	Area Under ROC Curve
BA	Binding Affinity
BFA	Brefeldin A
BH	Bleomycin Hydrolase
(Bi)LSTM	(Bidirectional) Long Short-Term Memory
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
CNN	Convolutional Neural Network
CTL	Cytotoxic T-Lymphocyte
EL	Eluted Ligand
ER	Endoplasmic Reticulum
ERAP1/ERAP2	Endoplasmic Reticulum Aminopeptidase 1/2
ESM	Evolutionary Scale Modelling
ESS	Effective Sample Size
GBM	Glioblastoma Multiforme
HLA	Human Leukocyte Antigen

IEDB	Immune Epitope Database
IFNg	Interferon Gamma
IGF	Insulin-like Growth Factor
LAP	Leucine Aminopeptidase
MAGE	Melanoma Antigen Gene
MAP	Maximum A Posteriori
MCMC	Markov Chain Monte Carlo
MFI	Mean Fluorescence Intensity
MHC(-I)	Major Histocompatability Complex (I)
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
NBD	Non-Binding Domain
NGS	Next-Generation Sequencing
NK	Natural Killer (Cell)
ODE	Ordinary Differential Equation
ORF	Open Reading Frame
PAM	Point Accepted Mutation
PCA	Principal Component Analysis
PINTS	Probabilistic Inference on Noisy Time Series
PK/PD	Pharmacokinetics and Pharmacodynamics
PLC	Peptide Loading Complex
pMHC	Peptide Major Histocompatability Complex
PR	Precision-Recall
PSA	Puromycin Sensitive Aminopeptidase
PWM	Positional Weighted Matrix
ROC	Receiver Operating Characteristic
SARS	Severe Acute Respiratory Syndrome
SMM	Stabilised Matrix Method
SNP	Single Nucleotide Polymorphism

SSE	Sum of Squared Errors
SVM	Support Vector Machine
SVR	Support Vector Regression
TAP	Transporter associated with Antigen Processing
TCR	T-Cell Receptor
TMD	Trans-Membrane Domain
TPPII	Tripeptidyl Peptidase II
WT	Wild Type
XNES	Exponential Natural Evolution Strategy

Chapter 1

Introduction

Vaccination is a critical public health strategy designed to protect individuals and populations from infectious diseases. Its importance was emphasised recently during the race to develop a vaccine against the SARS-CoV-2 virus during the COVID-19 pandemic. The resulting vaccines, developed at record-breaking speed, were crucial in reducing rates of infection, severity of disease, and mortality [174].

In recent years, vaccination has emerged as a promising strategy in oncology, offering both preventive and therapeutic strategies against the insidious nature of cancer. In the early 2010s, cancer vaccines were developed targeting shared tumour associated antigens — proteins expressed by cancer cells but not significantly by healthy cells. These antigens, such as MAGE-A3 in melanoma, were used to stimulate the patient's immune system to recognise and attack tumours expressing these markers [159]. Despite the initial promise, the effectiveness of vaccines targeting shared antigens was limited by the heterogeneity of cancer cells, their ability to evade immune detection, and immune self-tolerance mechanisms.

This has led to a shift towards the exploration of the use of patient-specific ('personalised') neoantigens for immunotherapies [134]. The advent of next-generation sequencing (NGS) and improved computational algorithms has opened the door to efficient identification and prioritisation of neoantigens on a patient-by-patient level. The feasibility of such an approach has already been demonstrated by Ott et al. in melanoma patients, where vaccination with neoantigen peptides resulted in sustained tumour regression [121].

The COVID-19 pandemic also provided a window of opportunity for the testing of

advanced vaccine delivery platforms. This was most successfully shown by the safety and efficacy of the mRNA-based vaccines developed by Pfizer and Moderna, and the 'Oxford vaccine' (ChAdOx1 nCoV-19) deployed around the world [11, 127, 161].

However, a major limitation in the development of new immunotherapies still exists in the selection of appropriate antigen targets for vaccines. This is particularly problematic in the case of cancer vaccines, where NGS might identify a long list of neoantigens. In order to choose the most appropriate targets, it is vital that we can swiftly and accurately predict the ability of neoantigens or pathogenic peptides to initiate an immune response that leads to activation of cytotoxic T-lymphocytes (CTLs). This property is known as *immunogenicity*.

1.1 Research motivation

Computational methods have been developed in the past 20 years in order to predict neoantigen presentation and immunogenicity. Whilst the accurate *in silico* prediction of immunogenicity is still viewed as an elusive problem, limited by the availability of training data [92], it is widely accepted that modern predictors of antigen presentation are high-performing and operating towards the upper limit of attainable accuracy [118, 130, 173].

In this thesis, we propose that existing models are effectively predicting the *presence* of presented peptide-MHC (pMHC) complexes, rather than their *abundance*. This distinction arises from the lack of quantitative measurements of absolute pMHC presentation available for use in machine learning. Instead, modellers have relied on eluted ligands (EL) identified through immunopeptidomics studies. The techniques employed in these studies (as detailed in Section 1.3.6.1) allow for the identification of pMHC complexes, but not their quantification. Consequently, the resulting datasets consist solely of pMHC complexes identified from various cell lines (positively labeled), supplemented by random natural peptides not found among the eluted ligands and thus assumed to be poorly presented by the cell line (negatively labeled).

However, antigen presentation is a spectrum and hence should not be treated as a binary classification problem. Epitopes can be presented in quantities between one and tens-of-thousands, as shown for influenza A epitopes by Wu et al. in a dendritic cell line (reproduced in Figure 1.1) [168]. Both of the H2-Db restricted epitopes $NP_{366-374}$ (arithmetic mean of 3,871 pMHCs across 3 replicates) and NP_{55-63} (arithmetic mean of 4 pMHCs across 3 replicates) would be assigned a positive label in the

training or testing of a predictive model such as *NetMHCpan-4.1*, despite an over 500-fold difference in raw presentation.

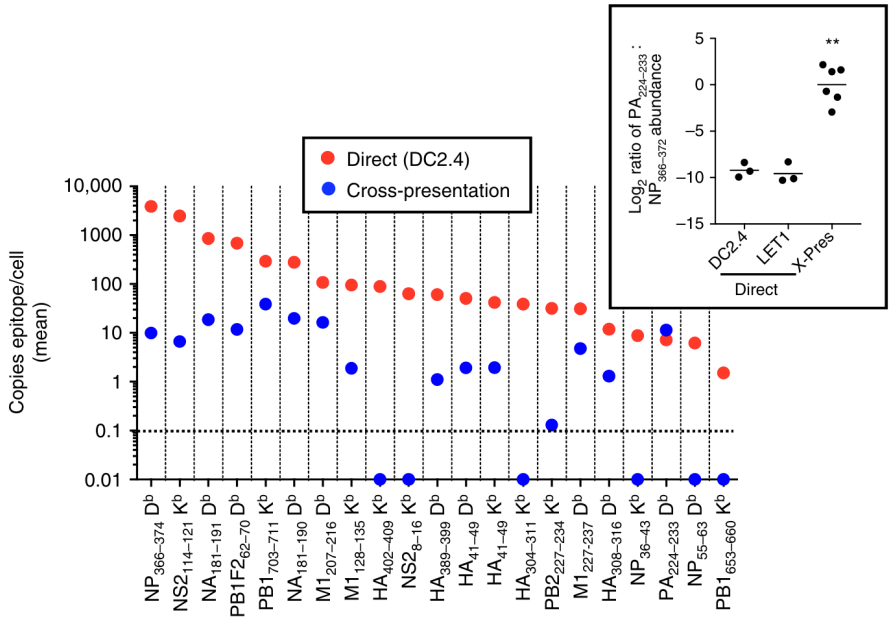


Figure 1.1. Absolute quantification of presentation of influenza A peptides, showing the range of pMHC abundance across 21 class I epitopes. Reproduced from Wu et al. [168].

Accordingly, predictive methods using this binary training data perform extremely well at predicting antigen presence but are often unable to estimate antigen abundance (as exemplified by correlation strengths between their outputs and the data from Figure 1.1, shown in Table 1.1). This has significant consequences for the prediction of immunogenicity because the CD8+ T-cell response magnitude has been shown to correlate positively with the pMHC abundance of the cognate antigen [168]. Hence, we conclude that a data-driven approach is not viable for the prediction of raw pMHC abundance and an alternative strategy is needed.

Predictor	**H2-Kb**		**H2-Db**	
	R_p	R_s	R_p	R_s
BigMHC EL [10]	-0.148	0.479	-0.194	-0.327
NetMHCpan-4.1 [130]	0.249	0.297	0.542	0.009
MHCflurry-2.0 [118]	0.367	0.527	0.220	-0.227

Table 1.1. Correlation between outputs of 3 antigen presentation predictors from the literature and the absolute quantification of direct presentation from the Wu et al. dataset. Performance is separated by restricting MHC-I allele. Correlation is given in terms of the Pearson correlation coefficient, R_p, and the Spearman correlation coefficient, R_s.

The antigen processing and presentation pathway has been well-characterised since the initial discovery of the major histocompatability (MHC) complex in mice by Peter Gorer in 1936 [68]. Furthermore, the specificities of the various enzymes and transport proteins involved in the pathway have been investigated *in vitro*, with plenty of measurements of their activities and binding affinities made publicly available [6, 20, 156]. This inspired us to investigate whether a mechanistic, immunology-driven approach to the prediction of antigen presentation might prove more effective than a data-driven one.

In this thesis, we develop a computational model of the class I antigen processing and presentation pathway using a hybrid of machine learning and systems biology. We subsequently use this to train a novel predictor of CD8+ immunogenicity. In the remainder of this chapter, we present a background of the relevant immunology and computational methods to provide the reader with context for the subsequent research.

1.2 Biological review

1.2.1 The adaptive immune system

The vertebrate immune system has two branches: the *innate* and the *adaptive* immune system. The innate immune system is the first line of defence against pathogens, characterised by its rapid response and lack of antigen specificity. It comprises multiple subsets of leukocytes (white blood cells), including macrophages, neutrophils, natural killer (NK) cells, and dendritic cells (shown in Figure 1.2). Macrophages and neutrophils are primarily responsible for phagocytosing pathogens, while NK cells play a crucial role in identifying and destroying virally infected cells and tumour cells in response to the changes in expression of surface ligands (including peptide-MHC-I complexes), which are sensed via receptors including killer cell immunoglobulin-like receptors (KIRs). Dendritic cells, on the other hand, are specialised cells for antigen presentation, bridging the innate and adaptive immune responses by activating T cells.

Innate leukocytes detect the presence of infection or other "danger" situations through pattern recognition receptors (PRRs) that recognise pathogen-associated molecular patterns (PAMPs) or damage-associated molecular patterns (DAMPs). Upon activation, members of the innate system can express signaling molecules,

such as chemokines and cytokines, that increase cell proliferation, promote inflammation, or drive the recruitment of other members of the innate or adaptive immune system to the site of infection. Through the co-evolution of the innate system and pathogens over many hundreds of millions of years, some pathogens have evolved to evade the innate system. Immune evasion methods include alteration of their surface proteins, inhibiting these PRR signaling pathways, or hiding within other cells, thereby avoiding detection. In cases such as these where the innate immune system is insufficient to fully eliminate a threat, the adaptive immune system is called upon.

Adaptive immunity is a targeted and sophisticated immune defence, mediated by two major classes of cell: T-cells (named for their maturation in the thymus) and B-cells (which mature in the bone marrow) — both members of a family of cells known as lymphocytes (shown in Figure 1.2). The receptors on these cells recognise (almost) a unique molecule with high affinity. This complementary molecule is known as its *cognate antigen*.

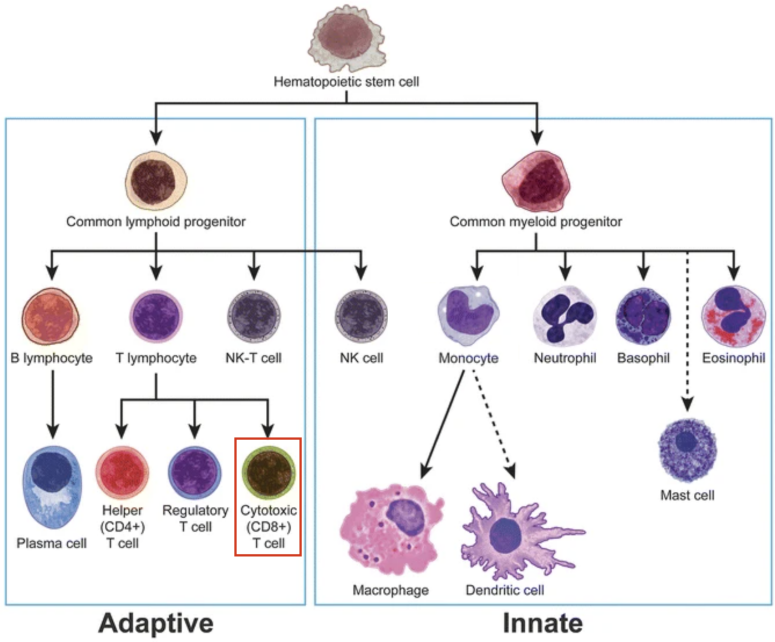


Figure 1.2. Differentiation of white blood cells from pluripotent stem cell. The focus of this thesis is on CD8+ T-cells, highlighted in red box. Reproduced from [123]

B-cells recognise antigens through membrane-bound immunoglobulins on their cell surface called B-cell receptors (BCRs). These BCRs bind directly to specific antigens in their native form, resulting in B-cell activation. Activated B-cells produce and secrete immunoglobulins known as *antibodies* with a variety of functions. Antibodies can 'decorate' invaders, thus preventing viruses from entering cells or impeding their function, can clump bacteria and viruses together, or can facilitate the binding of

natural killer (NK) cells to target cells, amongst other purposes. There are three major types of T lymphocyte, classified according to their surface receptors: (i) helper T-cells (typically expressing the co-receptor CD4), (ii) regulatory T-cells (also typically expressing the co-receptor CD4, along with the transcription factor FOXP3), and (iii) cytotoxic T-cells (typically expressing the co-receptor CD8). Helper T-cells express cytokines and engage in receptor-ligand interactions that activate the functions of other cells, such as B-cell production of antibodies and macrophage killing of engulfed pathogens. Regulatory T-cells suppress the activity of other lymphocytes in order to limit the possible damage of immune responses. Cytotoxic T-cells kill cancer cells or cells that are infected with viruses through the release of cytotoxic granules and engagement of death receptors.

Ordinarily, non-activated (naïve) lymphocytes exhibit little activity and cells recognising any given antigen are present in low numbers. Upon infection by a pathogen, members of the innate immune system known as antigen presenting cells (APCs) present antigen bound to specialised cell-surface glycoproteins called major histocompatability complex (MHC) molecules to naive T-cells in the secondary lymphoid tissues (the lymph nodes and the spleen). These APCs either take up antigen from the infection site or present antigen generated within the APC itself.

Extracellular antigens are typically processed and presented on MHC class II (MHC-II) molecules to CD4+ T cells, which helps to activate helper T cells. Alternatively, antigens derived from proteins synthesized within the APC, such as viral proteins, are presented on MHC class I (MHC-I) molecules to CD8+ T cells, a process known as direct MHC-I presentation. In addition, APCs can also engage in cross-presentation, where extracellular antigens are presented on MHC-I molecules, allowing CD8+ T cells to recognise and respond to pathogens or tumor cells that do not directly infect the APC.

Providing sufficient stimulation occurs of the cell receptors, this causes the T- and B-cells to proliferate and differentiate into effector or memory cells. Effector cells specific to the particular antigen are present at much higher frequencies and exhibit increased expression of co-receptors and other activation markers, improving their ability to effectively respond to the pathogen. Effector cytotoxic T-cells, in particular, become highly cytotoxic, with enhanced capabilities to kill infected or malignant cells through the production of cytotoxic molecules and the expression of death-inducing ligands. Although these effector cells play a crucial role in the immune response, their lifespan is short, and once the infection passes, they die. This leaves behind the memory cell population, which is able to rapidly proliferate into even greater numbers

of effector and memory cells in future should the cognate antigen be presented once again.

In this thesis, we focus on class I antigen processing and CD8+ T-cell activation. We now discuss these processes in greater detail.

1.2.2 Class I antigen processing and direct presentation

In this section and the remainder of this thesis, we focus exclusively on direct presentation. This focus is driven primarily by the critical role of direct presentation in predicting neoantigen presentation by cancer cells, which is essential for the effective targeting of tumor cells by CD8+ T-cells. Additionally, direct presentation plays a key role in the priming of CD8+ T cells by professional APCs following the uptake of vaccine mRNA. While cross-presentation also contributes significantly by enabling the presentation of extracellular antigens from non-APC cells, direct presentation remains central to the processes under investigation, particularly in the context of cancer immunotherapy and vaccine responses.

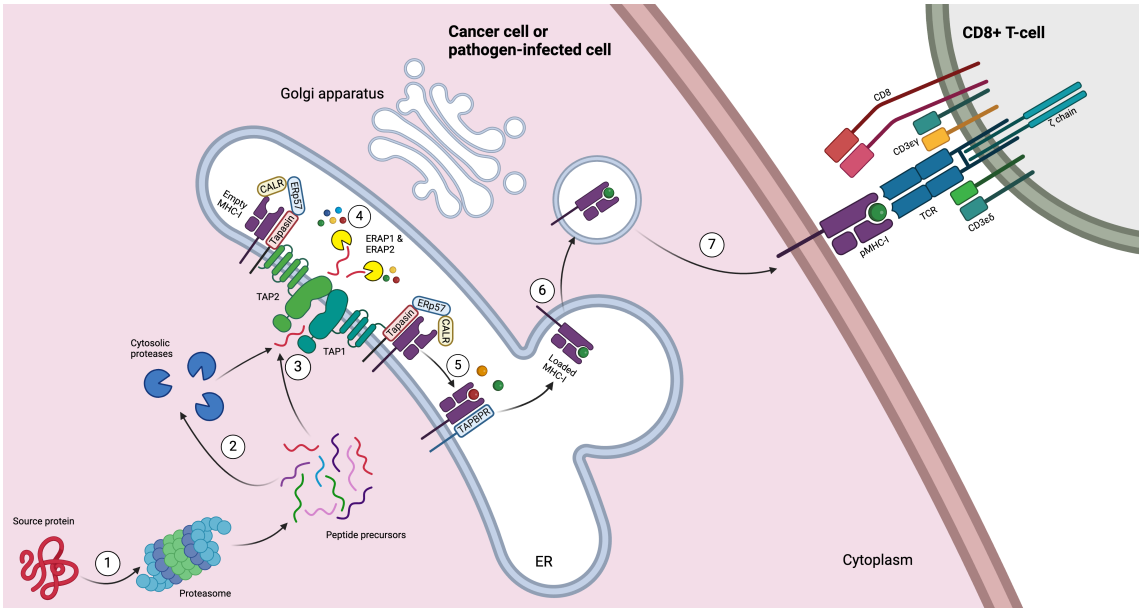


Figure 1.3. A schematic of the direct presentation pathway, produced using BioRender. The key steps in the pathway are numbered from 1 to 7.

A series of processes must occur in order for a peptide to be extracted from its source protein and presented to the naïve CD8+ T-cells via the direct presentation pathway. The main steps are shown in the schematic in Figure 1.3. It should also be noted that some peptides are generated in the endoplasmic reticulum, not the

cytosol (e.g. from protein signal sequences). These peptides are not shown in the diagram and are not considered in the scope of this thesis.

1.2.2.1 Proteasomal cleavage

Proteins in cells are continually being degraded and replaced by newly synthesised proteins to maintain a steady-state abundance of each endogenous protein. If we denote the synthesis rate of a protein by g and the degradation rate of a protein by d, then the steady state abundance of protein is given by $P := g/d$. Hence, proteins with high cellular abundance either have a high synthesis rate, a low degradation rate, or both.

Of greater interest in studying the proteasome is the protein turnover. This is given by the rate of protein degradation $d \times P$, which is equal to g at steady-state. The majority of cytosolic protein degradation is carried out by a large protease complex called the proteasome (Step 1). In addition to the degradation of these functional proteins, non-functional proteins resulting from errors in protein synthesis (e.g. aberrant translation and defective ribosomal products, or DRiPs) are also targeted to the proteasome.

The constitutive proteasome is composed of a 20S catalytic core and two 19S regulatory caps, one at each end, as indicated in Figure 1.3. During an immune response, a subset of cytokines known as interferons can alter the expression of key genes involved in the antigen processing pathway. If the surface receptors of an antigen presenting cell are stimulated by interferon-gamma (IFNg), the composition of the the cell's 20S subunit can sometimes be changed, forming an alternative structure of the constitutive proteasome known as the immunoproteasome. The immunoproteasome has been shown to have different specificity and activity to the constitutive proteasome [77, 82].

Proteins in the cytosol are labelled for degradation by the attachment of a chain of several ubiquitin molecules, a process called ubiquitination. This chain of ubiquitins is recognised by the 19S cap of the proteasome, which unfolds the protein so that it might be fed into the proteasome's catalytic 20S core. Here, the protein chain is cleaved into short peptides which are subsequently released into the cytosol. Two thirds of these peptides are below the minimum length typically required for loading to MHC-I [84]. The remaining products include MHC-I epitopes and N-terminally extended epitope precursors. Minimal subsequent C-terminus processing of these peptides is believed to occur. Hence, the proteasome is currently believed to be the

protease predominantly responsible for the generation of the C-terminus of MHC class I epitopes [166].

A protein may enter the proteasome from its N- or C-terminus. The direction of substrate entry has been shown to affect the peptides the proteasome generates. The preferred direction appears to correlate with the force required to unfold the corresponding terminus, as revealed by molecular dynamics simulations [18]. Inhibition of the preferred terminus was also found to have a significant detrimental effect on downstream class I antigen presentation, showing that the proteasome plays a key role in epitope generation and that the effect of direction of substrate entry is integral to this process.

1.2.2.2 Cytosolic aminopeptidases

The products of the proteasome, released into the cytosol, include many peptides that are too long to bind with high affinity to MHC-I. A collection of aminopeptidases in the cytosol play an important role in trimming the N-terminus of these elongated peptides (Step 2), including puromycin sensitive aminopeptidase (PSA), bleomycin hydrolase (BH), leucine aminopeptidase (LAP), and tripeptidyl peptidase II (TPPII). Knockouts of LAP, BH and PSA in mice have not shown a significant difference in peptide trimming or the presentation of epitopes, suggesting that there may be considerable functional redundancy between these aminopeptidases [150–152]. Both the individual and collective specificities of these enzymes show dependence on the substrate's N-terminus [6, 139].

Other proteases in the cytosol rapidly destroy short peptides, contributing to an extremely short average half-life of only a few seconds [96, 129]. Hence, to stand a chance of being presented, it is vital that peptides are removed from this harsh environment efficiently. This is carried out by the transporter associated with antigen processing (TAP).

1.2.2.3 TAP translocation

TAP is a heterodimeric complex of the TAP1 and TAP2 proteins. The transporter is embedded in the membrane of the ER and is responsible for the translocation of peptides of lengths between 8 and 16 residues from the cytosol into the ER lumen (Step 3) [157]. The process by which a peptide is translocated can be broken down into 4 stages:

1. **Initial state**: In its resting state, TAP adopts an inward-facing conformation, in which the trans-membrane domains (TMDs) of TAP1 and TAP2 block peptides from diffusing into the ER lumen.

2. **Peptide and ATP binding**: A peptide in the cytosol binds to the two TMDs. Concurrently, ATP molecules bind to each of the non-binding domains (NBDs), a process independent of peptide binding [2, 3].

3. **Conformational change and peptide release**: The binding of the peptide induces an allosteric change in the structure of TAP, bringing the NBDs together and reconfiguring the TAP to the outward-facing state. This releases the peptide into the ER lumen [3].

4. **Resetting the transporter**: The bound ATP is hydrolysed to ADP, leading to the destabilisation and subsequent dissociation of the NBD dimer. This change resets TAP to its original inward-facing state, ready to initiate another cycle of peptide transport.

Research has shown that the translocation efficiency of peptides by TAP is directly related to their binding affinity [27]. This correlation suggests that the association between the peptide and TAP constitutes the rate-limiting step in the peptide translocation process. Consequently, variations in peptide affinity can significantly influence the efficiency of antigen presentation, as efficient TAP translocation is generally required for peptide loading onto MHC class I molecules [156] (with the exception of a subset of peptides that are not dependent on TAP translocation, e.g. signal sequence-derived peptides). The importance of efficient TAP translocation can be further seen in the impact of TAP transport inhibition *in vivo*. An example of this is the competitive inhibition of TAP by the ICP47 protein following infection by the herpes simplex virus (HSV). This has been shown to significantly reduce peptide presentation, particularly in HLA-A and HLA-C, potentially resulting in a decrease in CTL recognition [119, 142].

1.2.2.4 ERAP1 and ERAP2 trimming

Due to TAP's broad length specificity, after translocation to the ER, peptides often exceed the optimal length of 8 to 10 amino acids required for efficient binding to the MHC-I groove. ERAP1 and ERAP2 are two aminopeptidases that play vital roles in this peptide processing (Step 4), exhibiting high substrate specificity that is essential for the generation of peptides that are compatible with MHC-I molecules.

ERAP1 has a unique mechanism of action, binding peptides at their C-terminus within a hydrophobic pocket while enzymatic cleavage occurs from the N-terminus at the active site [35]. This effectively serves as a 'molecular ruler', imposing a minimum length of approximately 8 amino acids to span from the hydrophobic pocket to the active site. Conversely, peptides above 16 amino acids do not bind effectively due to steric hindrance caused by folding in the internal cavity. The specificity of the hydrophobic pocket for C-terminal residues is highlighted by its preference for amino acids like leucine and valine [35].

In contrast, ERAP2 does not recognise the peptide C-terminus, so has a lack of C-terminal specificity [112]. This distinction from ERAP1 highlights the complementary nature of the two aminopeptidases. ERAP2 is particularly adept at cleaving peptides of length 9 residues or shorter. Because of its length specificity, in addition to the generation of MHC-I epitopes from precursors, ERAP2 is also theorised to enhance ERAP1 trimming efficiency by removing short peptides (<8 amino acids) that may act as competitive inhibitors [45].

ERAP1 and ERAP2 have differing substrate specificity at the N-terminus. ERAP1 shows a preference for hydrophobic or bulky residues such as leucine and methionine, whereas ERAP2 displays an affinity for basic residues, such as lysine and arginine [72, 137]. The complementary nature of ERAP1 and ERAP2's substrate specificities ensures that a diverse range of peptides can be processed in the ER. However, the extent to which the enzymes act in concert is not well understood. ERAP1 and ERAP2 have been observed to form heterodimers *in vivo*, increasing the efficiency of ERAP1 [137]. However, the enzymes are predominantly found in the monomeric form in cells, suggesting that these dimers may not contribute significantly to epitope generation.

Researchers have disagreed over the primary state of the peptide substrates of ERAP1 *in vivo*. One school of thought was that MHC-I might act as a template, with bound peptides being trimmed down to appropriate length by ERAP1 [23, 122]. However, recent studies have suggested that this is not likely to be a common occurrence, with free peptide being the most prevalent substrate for ERAP1 [111].

Research has predominantly focused on ERAP1 compared to ERAP2, possibly because ERAP2 is sometimes considered supplementary to ERAP1 .This notion is supported by evidence in the literature to suggest that ERAP1 is essential for the generation of immunogenic melanoma epitopes, but that inhibition of ERAP2 has minimal effect on the generation of these epitopes [147]. However, ERAP2 has been genetically associated with various diseases, including the Black Death [85]. This

implies a key role for ERAP2 in the cellular immune response which is yet to be fully understood.

Despite playing a key role in the generation of certain epitopes, it has also been shown that ERAP1 inhibition/knockdown can increase the presentation of high affinity 9-12mers, suggesting that ERAP1 activity can be destructive for many potential epitopes [86]. This led to a view that ERAP1 might be promising target for immunotherapies.

1.2.2.5 Peptide loading to MHC-I

Following the successful translocation of peptides into the ER by TAP, the peptides are loaded on to MHC-I (Step 5). This process involves a multi-component assembly called the peptide loading complex (PLC), which includes TAP, tapasin, calreticulin, ERp57, and the MHC-I molecules themselves. Each protein plays a specific role in ensuring efficient loading:

- **TAP** functions not only as the peptide transporter but also as the structural foundation for the PLC. Tapasin binds via its transmembrane region to TAP, facilitating the efficient transfer of peptides from TAP to the MHC-I molecule. This minimises the diffusion of peptides within the ER lumen, thus enhancing the efficiency of peptide loading.

- **Calreticulin** and **ERp57** form a chaperone complex with tapasin, contributing to the structural integrity of the PLC. Calreticulin binds to the heavy chain of MHC-I, assisting in the proper folding and retention of MHC-I until suitable peptides are presented for binding. ERp57 is bound to tapasin and plays a role in maintaining the correct conformation of tapasin.

- **Tapasin** is a critical component of the PLC which enhances the efficiency of peptide loading by stabilising the MHC-I molecule in an open conformation that is more receptive to peptide binding. It facilitates the release of low-affinity peptides from the MHC-I binding groove and enhances sampling of other available peptides, thus improving turnover and peptide filtering [167].

- **MHC-I** is composed of a heavy chain and β2-microglobulin and is the final recipient of the peptides. Following successful binding, the MHC-I dissociates from the PLC and egresses to the antigen presenting cell surface via the Golgi apparatus (Step 6), whereupon the pMHC complex will eventually dissociate, with the empty MHC-I typically being internalised and recycled (Step 7). The half-life of the pMHC complex depends on the affinity and stability of the bound

peptide. This property depends on the peptide sequence and the MHC-I allele it is bound to.

Although not part of the peptide loading complex, the chaperone molecule TAPBPR plays a significant role in ensuring that the repertoire of peptides presented on the cell surface primarily consists of stable pMHC complexes. It achieves this by facilitating the exchange of lower affinity peptides, previously bound to MHC-I, with higher affinity peptides.

MHC-I is among the most polymorphic genes in the human genome. This polymorphism occurs primarily in the peptide binding-groove, dramatically altering the set of peptides that can be bound and presented to T-cells. This diversity ensures that different individuals in a population can present a wide range of pathogenic peptides to T-cells, increasing the likelihood that at least some individuals can effectively respond to a novel pathogenic threat. Humans express 3 classical MHC-I genes: human leukocyte antigen (HLA)-A, -B, and -C. This significantly broadens the range of pathogens that the human immune system can respond to by increasing the likelihood that a pathogen-derived product will bind stably to at least one of the expressed MHC-I proteins.

As well as affecting the specificity of the binding groove, MHC-I alleles also have different levels of sensitivity to the presence of tapasin — a phenomenon known as *tapasin dependence*. Highly tapasin dependent alleles are characterised by inefficient peptide presentation in the absence of tapasin [14, 76]. In the presence of tapasin, these alleles present a greater abundance of strong binders and a lower level of weak binders than tapasin independent alleles, decreasing the breadth of the peptide repertoire.

1.2.3 CD8+ T-cell activation

Dendritic cells are considered to be the primary cells involved in the activation of CD8+ T-cells. Following detection of PAMPs and DAMPs by the dendritic cell PRRs (e.g. toll-like receptors), and stimulation by secreted cytokines, the dendritic cell undergoes maturation. This results in a change in the dendritic cell's proteins, function and morphology in order to better prepare it for T-cell activation, including:

1. Upregulation of MHC molecules, co-stimulatory molecules (e.g. CD80 and CD86), and chemokine receptors (e.g. CCR7).

2. Decreased antigen uptake through phagocytosis or endocytosis.

3. Increased cytokine production to drive the differentiation and activation of T-cells.

4. Increased surface area through the development of longer and more numerous dendrites, allowing for more efficient interaction with T-cells.

Upregulation of CCR7 causes the dendritic cell to respond to chemokines and migrate towards the lymph nodes, where they reside for a few days until they die. Naïve CD8+ T-cells randomly move around the lymph node paracortex, encountering APCs as they do. Most of these interactions are transient. However, when a T-cell encounters an APC presenting its cognate antigen, ligation occurs between a pMHC complex and a TCR, forming an immunological synapse. Large, transmembrane phosphatases on the T-cell surface (e.g. CD35 and CD148, which inhibit TCR signalling) are physically excluded from the close-contact zone between the T-cell and APC due to its narrow size of a few nanometers. Conversely, smaller molecules like TCRs and CD4/CD8 co-receptors are enriched in this zone. This is known as the kinetic segregation model and explains how activation can occur despite the presence of inhibitory mechanisms that would normally prevent activation. If a threshold signal is reached during this process, the T-cell becomes activated. Even antigens with lower affinity can activate T-cells through the integration of many signals resulting from interactions with a lower avidity.

1.3 Mathematical modelling review

1.3.1 Mathematical modelling in biology

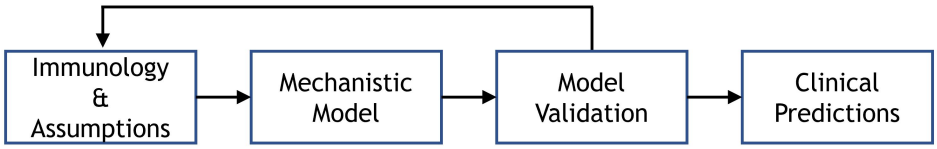


Figure 1.4. A pipeline showing the iterative development process for mechanistic models.

Mathematical models can be broken down into two main categories: mechanistic models and machine learning. Mechanistic models are developed using knowledge or assumptions about the underlying biology, with a focus on parsimony to avoid over-fitting. This was requirement for simplicity was famously advocated by the physicist Enrico Fermi, who claimed 'with four parameters I can fit an elephant, and with five I can make him wiggle his trunk' [51]. Effective mechanistic modelling takes

a complicated biological concept and represents it with the simplest mathematical description that produces predictions consistent with the available data. This often follows an iterative procedure (see Figure 1.4), starting with a minimal model and adding in additional features and complexity until the model is consistent with available validation data.

A prominent and successful example of early mechanistic modelling can be seen in the modelling of the initiation and propagation of action potentials in the squid giant axon by Hodgkin and Huxley in 1952 [75]. Hodgkin and Huxley showed that a system of just 4 ordinary differential equations (ODEs) could accurately reproduce their experimental observations from voltage clamping of the axon. This representation of the complicated underlying biology gave scientists an improved understanding of how neurons work whilst satisfying the simplicity so passionately advocated for by Fermi.

In the remainder of the 20th century and early 21st century, computational biologists used mechanistic modelling to: predict the absorption, distribution, metabolism, and excretion of drugs through pharmacokinetic and pharmacodynamic (PKPD) models [26]; model tumour growth and infiltration dynamics using reaction-diffusion models [65]; and elucidate the limitations and causes of failure from clinical trial data using agent based models (ABMs) [24, 25].

Machine learning[1], on the other hand, relies on the availability of abundant data rather than necessitating an understanding of the underlying mechanism. Large datasets can be used to train regressors or classifiers, the complexity of which is a choice made by the modeller with consideration to the bias-variance trade-off. Without being given information about the underlying biology, the model learns how the input features map to the outputs, thus implicitly capturing statistical relationships and correlations between these features caused by the underlying mechanism. Machine learning has been used biology in recent years to great effect for: drug discovery and development; cancer detection from imaging data using convolutional neural networks (CNNs) [9]; and prediction of protein structure from sequence using large language models such as DeepMind's *AlphaFold* [28].

Mechanistic modelling and machine learning have strengths and weaknesses relative to one another. Mechanistic modelling offers the advantage of providing a detailed and interpretable representation of systems based on established physical, chemical, or biological principles, making it valuable for understanding causal relationships and

[1] In this thesis, unless specified, we use the term 'machine learning' to refer specifically to supervised learning.

revealing underlying mechanisms. These models often require less data for development compared to machine learning models, as they are grounded in prior knowledge of the system rather than relying solely on large datasets for pattern recognition. Additionally, mechanistic models can make reliable predictions under new conditions if the underlying principles are well understood. However, developing these models can be time-consuming and complex, requiring deep domain knowledge and careful parameter estimation. Moreover, they are sensitive to the assumptions made during their development, which can limit their accuracy if the mechanisms are not fully captured or are oversimplified.

Machine learning, on the other hand, typically requires vast quantities of data for model training. However, it is often not the volume of data that is prohibitive, but the nature of the data. Training data should ideally be diverse so that the model development is not biased towards a certain characteristic, but in some cases this is not feasible.

For example, the majority of T-cell assays deposited in the Immune Epitope Database (IEDB) correspond to the HLA-A*02:01 allele since this is the most common HLA type in Western populations. With the training data dominated by one HLA allotype, immunogenicity predictors trained on these datasets are skewed towards this allele [29] and an internal comparison of publicly available algorithms found substantially better predictive performance on HLA-A*02:01 than on less common alleles (see Figure A.4 and Figure 7.11).

Another problem with machine learning stems from cases in which the negative class is difficult to observe. A prime example of this occurs in the prediction of proteasomal cleavage sites using datasets of known epitopes. The proteasome is assumed to be responsible for generating the C-termini of these epitopes, so they can be used as positive examples of cleavage sites. However, peptides with poorly cleaved C-termini are generated inefficiently in the cytosol, so are not presented on the cell surface. Modellers therefore have to make assumptions in order to generate negative examples for model training, introducing biases into their models in the process [115, 165].

Despite these limitations, machine learning is a powerful technique and, where sufficient training data of appropriate composition is available, should be able to emulate or exceed the predictive performance of mechanistic modelling, especially when the underlying mechanism is not fully understood. However, in situations where machine learning matches or exceeds mechanistic modelling's performance, mechanistic modelling can still offer valuable insight into underlying mechanism of action (which is not possible through machine learning) and parameter values can be adjusted to simulate experiments into the effects of different conditions.

1.3.2 Machine learning with peptides

As we shall reveal in the forthcoming sections, the fields of antigen processing, antigen presentation and immunogenicity prediction have been dominated in the past 25 years by machine learning methods. Although the model design and output differs across these models, they all use peptide sequences as inputs. To provide a clear idea of the framework used to train machine learning models using peptide sequences, we briefly discuss the ways in which amino acids can be encoded.

The vast majority of machine learning methods require features to be in numerical format. When training models using peptide sequences, we must therefore convert (encode) amino acids into a numerical representation.

The simplest way to represent an amino acid in numerical format is to use a sparse encoding (also called a 'one-hot' encoding). The 20 canonical amino acids are each represented by a vector with a one at the index corresponding to an arbitrary ordering of the amino acids and zeros everywhere else. Hence, the amino acid alanine may be represented by $(1, 0, 0, \ldots, 0)$ and arginine by $(0, 1, 0, \ldots, 0)$. Although straightforward to implement, the sparse encoding does not provide the model with information about the similarities between different amino acids. This can be problematic, especially if the training dataset is small in size, because the model is likely to form erroneous predictions for unseen amino acids.

An alternative strategy is to encode amino acids based on relevant physicochemical properties, such as hydrophobicity, charge, or side chain volume. This can enable the trained model to generate predictions for unseen amino acid using relevant biochemical properties. However, the choice of properties and their scaling can significantly affect model performance.

Substitution matrices, such as BLOSUM62 or PAM250, derived from multiple sequence alignments, are widely used for encoding amino acids. In this encoding method, each amino acid is represented by a vector of length 20, usually consisting of log-odds ratios. These scores at each position in the vector reflect the estimated cost of substituting the original amino acid in the peptide with each of the other amino acids. This method provides a numerical representation that encapsulates evolutionary substitution preferences. However, these preferences may not hold in specific cases, particularly those where unusual evolutionary pressures are present.

Finally, the advent of large pre-trained protein language models, such as ProtT5 and ESM-2, presents opportunities to leverage transfer learning for novel applications [53, 101]. These models can accept peptides as input and return low-dimensional numerical vectors (*embeddings*) that are aware of both the position and sequence

context. However, this approach may not always be appropriate, especially when employing pseudosequences. Since these models are trained on datasets of physical protein sequences, inputting non-physically plausible sequences (with adjacent amino acids not seen in physical proteins) might lead to misleading embeddings.

The 4 families of amino acid representation are summarised in Figure 1.5.

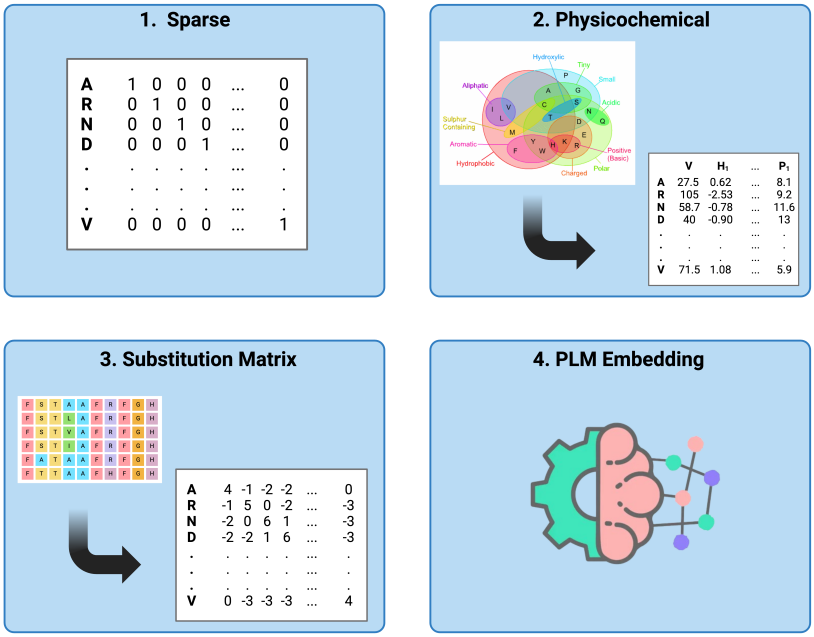


Figure 1.5. Comparison of the 4 most common strategies used to encode amino acids and peptides for use in machine learning.

We now explore the evolution and limitations of models of antigen processing and immunogenicity from the existing literature.

1.3.3 Prediction of proteasomal cleavage

1.3.3.1 A history of proteasomal cleavage prediction

Proteasomal cleavage prediction efforts began in 2000 with the Prediction Algorithm for Proteasomal Cleavages (*PAProC*) by Kuttler et al., predicting cleavage sites of human and yeast proteasomes [89]. The model used distance from previous cleavage and neighbouring amino acid composition to calculate the probability of a subsequent cleavage. Kesmir et al. then combined *in vitro* data with natural MHC-I ligands to vastly increase the data available for the training of their neural network, *NetChop* [83]. The C-termini of the ligands were assumed to have been generated by proteasomal cleavage, so represented positive examples of cleavage sites. Conversely,

internal residues in the ligands were assumed to have a low probability of cleavage, so were treated as negatively labelled training data (decoy samples). Each training point was encoded using a window of 18 amino acids, centered on the cleavage site, with amino acids represented by a sparse encoding.

The *NetChop* algorithm was then enhanced using an ensemble of neural networks, giving us the *NetChop 3.1* algorithm [115]. Around the same time, Bhasin and Raghava proposed *Pcleavage*, a cleavage predictor based on support vector machines (SVM), finding that this method outperformed other classifiers [19].

Following a brief hiatus in the field, the *Pepsickle* algorithm was published in 2021 by Weeder et al., using a similar framework to *NetChop 3.1*, but benefiting from an increased volume of *in vitro* and MHC-I ligand data [165]. Dorigatti et al. also published *PUUPL*, the novel contribution of which was to treat decoy samples as unlabelled, rather than negative [49]. The *PUUPL* authors used a semi-supervised learning method to assign pseudo-labels to the most confidently classified decoys at each iteration of the training process, thus overcoming some of the complications described in Section 1.3.1. Finally, Ziegler et al. published a review of proteasomal cleavage methods in 2023 in which they trained a bidirectional long short-term memory (BiLSTM) network and showed superior performance compared to the aforementioned methods across a benchmarking study [175].

1.3.3.2 Challenges to proteasomal cleavage prediction

The main challenges to proteasomal cleavage prediction can be broken down into two issues: (i) an insufficient supply of appropriate training data, and (ii) a lack of negative examples with which to balance classes in the available training dataset.

Kesmir et al.'s use of MHC-I ligand data resulted in only a modest 124% increase in the available training data, but by the time Weeder et al. trained *Pepsickle*, the addition of the publicly available IEDB had caused the available ligand data to dwarf the available *in vitro* data by a ratio of over 300 to 1. However, the use of ligand data introduces biases into the model training. For an MHC-I ligand to be presented, it must have successfully passed through the subsequent stages of the antigen processing pathway, so as a minimum must have sufficient level of TAP and MHC-I binding affinity. Both of these properties are heavily linked to the C-termini of the substrate, so their influence will likely bias the training set to overestimate cleavage of residues associated with favourable TAP and MHC-I binding.

Furthermore, proteasomal cleavage predictors also suffer from the lack of availability of negatively labelled training data. Whereas C-terminal cleavage is a necessary

condition for an MHC-I ligand to be presented, the presence of uncleaved internal residues in a ligand does not necessarily guarantee that these are unfavourable cleavage sites. Hence, anyone using ligand data must compromise between either using heavily class-imbalanced training data or using inaccurately labelled data, unless an approach like positive-unlabelled learning is used (e.g. in the *PUUPL* algorithm) [102]

1.3.4 Prediction of TAP translocation

1.3.4.1 History of TAP IC50 prediction

Multiple approaches have been published in the literature to predict the binding affinity (IC50) of peptides with human TAP. The first such method used artificial neural networks trained on a set of 381 different 9mer peptides of known binding affinity [44]. Peters et al. later proposed a stabilised matrix method (SMM), trained using a slightly larger set of IC50s from 430 different 9mers [126]. Peters et al. also suggested how their SMM might be used for N-terminally extended substrates by taking only the scores for the three N-terminal residues and the C-terminus.

Substantially improved predictive performance using support vector machines (SVM) was first reported by Bhasin and Raghava in 2003 with the *TAPPred* algorithm, trained on the same dataset as the SMM method [20]. Diez-Rivero et al. then expanded the training dataset to 613 9-mers, yielding a modest improvement with *TAPREG* ($R_s = 0.89$) compared to both SMM ($R_s = 0.87$) and *TAPPred* ($R_s = 0.67$, suggesting poor generalisation of this method to the new peptides) [48]. As with proteasomal cleavage prediction, interest in this field dwindled in the 2010s before reigniting in 2023 with the publication of *DeepTAP* — a recurrent neural network (RNN)-based method, trained on 868 peptides between 6 and 17 amino acids in length [172] ($R_s = 0.90$). This represented the first published model since the SMM method to form predictions for non 9-mer substrates. Whereas Peters et al. used just the N1-3 and C-terminal residues for longer peptides, *DeepTAP*. padded all peptides up to a length of 17 using the unknown amino acid, *X*.

1.3.4.2 Limitations of existing TAP IC50 predictors

Aside from the SMM method and *DeepTAP*, the TAP binding affinity predictors can all only form predictions for 9mers. This is clearly inadequate for studying the antigen processing pathway, since a large proportion of the proteasomal products (i.e. the pool of potential TAP substrates) are expected to be in the range of preferred TAP substrate lengths(8 to 16 amino acids), rather than exclusively 9mers [84].

Furthermore, all of the published models were trained using exclusively human TAP data, despite a number of studies into TAP specificity from mice and rats [31, 69]. This restricts the application of these models' predictions, particularly in the context of pre-clinical models, where a researcher might feasibly wish to predict the translocation efficiency of different peptides in mice or non-human primates.

1.3.5 Prediction of ERAP1 trimming

ERAP1 has a complicated substrate specificity due to its binding and active sites. Despite *in vitro* enzymatic assays and peptidomic analysis of ERAP1 expressing and knockout cell lines, only a single study attempting to predict ERAP1 trimming could be found in the literature, but at time of writing this model is still in peer review [8]. This model, *ERAMER*, uses a position weight matrix, constructed from experimental ERAP1 specificity studies [35, 55, 72], to calculate an ERAP1 trimming score for substrates between 9 and 16 residues in length. However, it is not clear how these scores can be rescaled or converted to meaningful enzyme kinetic parameters (e.g. k_{cat} or K_M) for integration into mechanistic models.

1.3.6 Prediction of MHC-I binding affinity

1.3.6.1 History of binding affinity prediction

Attempts to predict peptide-MHC binding affinity can be dated back to 1994 with the publication of the BIMAS algorithm to predict binding for 9mers to HLA-A2 [124]. Binding affinities from a set of 154 peptides were used to generate a matrix of 180 coefficients (20 amino acids x 9 positions). This method was later extended to other common HLA allotypes using the same approach.

In 2003, Nielsen et al. published an artificial neural network method, representing a huge step forward in binding affinity prediction [115]. This would become the first generation of the *NetMHC* family of algorithms. The authors' insight was to use a consensus prediction from two neural networks trained from sparse and BLOSUM50 encodings of the peptides. They also had access to a larger training set than BIMAS, with 528 binding affinities for 9mers with HLA-A2 available by this point in time.

In 2007, Nielsen et al. devised a strategy to form predictions for MHC-I alleles with limited training data, publishing *NetMHCpan* [114]. They represented the MHC-I sequence using a subset of the residues predicted to be directly involved in peptide binding. This shortened sequence was referred to as a *pseudosequence*. Nielsen et al.

combined data from across different alleles to train a single binding affinity predictor, once again using artificial neural networks. However, as well as the peptide sequence, they appended the associated pseudosequence to the network input features, thus enabling the network to learn the underlying biochemistry from an increased training set whilst still being able to tailor its predictions to the MHC-I allele. Nielsen et al. also discovered that adding random peptides into the training data with assumed weak binding affinity values reduced bias.

NetMHCpan has been a highly influential framework in the development of peptide-MHC binding affinity predictors since its publication. Over the years, its iterations have added new features and incorporated increasingly large datasets, notably including data from eluted ligands identified via mass spectrometry. To identify these peptides, the antigen presenting cells are lysed and the MHC-I molecules immunoprecipitated using complementary antibodies. The peptides are then eluted from the MHC-I molecules and often separated by liquid chromatography before being ionised and analysed by mass spectrometry. The measured mass to charge ratio (m/z) of the ionised peptides produces spectra which are subsequently matched to reference spectra using specialised software. The nature of this pipeline introduces certain biases into the set of identified ligands. Loss of peptides typically occurs during sample preparation due to dissociation of the peptide-MHC complex. This can result in lower affinity presented peptides not being identified. Furthermore, the theoretical spectra databases may be incomplete or biased, leading to relevant peptides being not identified or misidentified. However, despite these challenges, this method is high throughput, so significantly increases the training data available for training binding predictors.

In the training data for *NetMHCpan-4.0*, these eluted ligands, assumed to have a high binding affinity due to their presence as MHC-presented peptides, were uniformly assigned a maximal affinity [79]. Negative peptides, assumed to bind with insufficient affinity to be presented, were randomly sampled from the source antigens in the mass spectrometry data. In the most recent version, *NetMHCpan-4.1*, a motif deconvolution tool, *NNAlign_MA*, was introduced, allowing a vastly increased volume of mass spectrometry data to be used by mapping previously ambiguous eluted ligands to the (one of six) MHC-I alleles they were most likely bound to [130]. The addition of eluted ligand data has led to a modest improvement in binding affinity prediction over *NetMHCpan-3.0* [117].

1.3.6.2 Challenges to binding affinity prediction

The issues with binding affinity prediction closely mimic those mentioned when discussing proteasomal cleavage. Limited training data has led to the use of alternative types of data (eluted ligands) as a proxy for the variable of interest (binding affinity). Although the *NetMHCpan-4.1* training set contained 208,093 *in vitro* binding affinity measurements across 170 MHC-I alleles, this was dwarfed by the 663,767 eluted ligands in the training set, emphasising the incentive to use this type of data. However, the use of eluted ligands conflates the concepts of MHC-I binding and antigen processing. The assumption that random, non-eluted peptides from the source antigen must have a low binding affinity is not necessarily valid. Peptides may not be presented because of excessive or insufficient trimming by aminopeptidases, low TAP binding affinity, or inefficient production by the proteasome. Hence, assigning these random negatives a low affinity may lead to erroneous predictions.

Furthermore, a broad spectrum of binding affinities will likely exist amongst eluted ligands. Assigning them all a uniform high binding affinity might reduce the accuracy of binding affinity prediction amongst high affinity peptides. This issue is not widely recognised because binding affinity prediction has been reformulated as a classification problem since the integration of eluted ligand data, grouping high affinity peptides together into the positive class [118, 130]. However, evidence for this can be seen in a systematic benchmarking study by Zhao and Sher, who found weaker correlation between predicted and measured absolute binding affinity for strong binders [173].

1.3.7 Prediction of antigen presentation

1.3.7.1 History of antigen presentation prediction

Early approaches to predicting antigen presentation combined *in silico* predictions of MHC-I binding affinity with predictions of other components of the antigen processing pathway. In one such example, Tenzer et al. combined novel proteasomal cleavage scores with TAP and MHC-I binding affinity predictions in 2005, demonstrating that knowledge of the antigen processing pathway could be leveraged to improve predictive methods [146].

The advent of high-throughput pMHC identification using mass spectrometry led to a shift from this mechanism-inspired approach to a machine learning approach. The artificial neural networks used in *NetMHCpan-4.0* and *4.1* return two outputs: a binding affinity score and an eluted ligand score. The ANNs were trained to predict

binding affinity and classify eluted ligands simultaneously, improving performance [79, 130].

Similar strategies have been published elsewhere to use eluted ligand data to predict antigen presentation. In *MHCflurry-2.0*, O'Donnell et al. train both a binding affinity predictor and an antigen processing predictor using the top 2% of predicted binding affinities [118]. They include N- and C-terminal flanking regions on the peptide to steer the convolutional neural network to implicitly learn the rules of epitope generation from precursors, finding consistency with an independent dataset of proteasome-cleaved peptides. The binding affinity and antigen processing predictors were then combined using a logistic regression into a single predictor of antigen presentation.

Recent advances in machine learning methods have inspired new approaches to antigen presentation prediction. Albert et al. adopted a deep learning approach called *BigMHC-EL* to predict antigen presentation, utilising both the *NetMHCpan-4.1* and *MHCflurry-2.0* datasets, and proposing a novel MHC-I pseudosequence, showing impressive classification performance across a wide range of alleles [10].

1.3.7.2 Advantages of using immunopeptidomics to predict antigen presentation

The major advantage of using immunopeptidomics data to predict antigen presentation is that the data represents the overall output of the various steps occuring in the antigen processing pathway. For an eluted ligand to have been identified on the cell surface, it must have had its C-terminus generated by proteasomal cleavage, been translocated efficiently by TAP, and bound to MHC-I with sufficient affinity for the complex to egress to the cell surface before dissociating. Hence, the immunopeptidomics data includes signal from the underlying stages of the pathway and the total product of their interaction.

Immunopeptidomics datasets are also relatively prevalent in the literature and the method is high-throughput, giving a large number (approximately 600,000 in *NetMHCpan-4.1*) of pMHC pairs for use in training machine learning algorithms to predict antigen presentation.

1.3.7.3 Limitations of using immunopeptidomics to predict antigen presentation

The main drawbacks with using immunopeptidomics datasets for this task are linked to biases in the protocols used and the limitations of mass spectrometry. Low-affinity pMHC complexes are prone to dissociation during the immunoprecipitation process used to isolate pMHCs for mass spectrometry analysis as several washes are often required to remove non-specific peptides [90]. These low-affinity peptides are also generally present in a lower abundance than high-affinity ones, so may not be detected during mass spectrometry, depending on the sensitivity. Both of these problems can bias the immunopeptidomics datasets towards high-affinity pMHC complexes and result in some low-affinity pMHCs being falsely treated as negative samples in training datasets.

Furthermore, although the data provided by immunopeptidomics is far more abundant than the data available for studying the different stages of the antigen processing pathway (MHC-I/TAP binding affinity assays, proteasomal digestion assays, etc.), mass spectrometry is only a semi-quantitative technique and is typically unable to inform the user about how many copies of each pMHC were eluted from the cell surface. This leads to antigen presentation being treated as a binary classification problem rather than a regressive one: an eluted ligand is either labelled as having been identified ('positive') or not ('negative'). Hence, an efficiently-presented hypothetical pMHC complex with 10^4 copies on the cell surface would be labelled identically to a pMHC with only 1 copy on the cell surface. As a result, the outputs of these machine learning methods correspond to the likelihood that a particular peptide-MHC pairing would be eluted from a cell expressing the source protein and the MHC-I allele. These predictive methods could therefore more appropriately be described as predictors of antigen *presence*, not antigen *abundance*. As shown in Table 1.1, these algorithms performed poorly when tasked with predicting raw pMHC numbers, suggesting that their efficacy as classifiers does not translate into efficacy as regressors. However, more comprehensive benchmarking of their performance across a broad range of alleles is not possible due to limited quantitative pMHC data.

1.4 Prediction of CD8+ immunogenicity

Prediction of immunogenicity requires consideration of two criteria: (i) the level of presentation of the peptide, and (ii) the pMHC's ability to activate a naïve T-cell. Accordingly, the inputs to most immunogenicity predictors in the literature consist

of a prediction of antigen presentation coupled with a representation of the peptide sequence.

A simple example of this can be seen in *NetTepi* [153]. Trolle et al. take the weighted sum of predicted pMHC affinity and stability (as a simple proxy for antigen presentation) with a predicted T-cell propensity score. This propensity score is calculated using a study into enriched residues at different sites of known immunogenic epitopes [32].

A slightly more complex model can be seen in *PRIME-2.0* — a fully-connected neural network with one hidden layer, taking as its input:

1. Predicted presentation rank from the group's antigen presentation predictor, *MixMHCpred-2.2* [66].
2. Peptide length encoding.
3. Frequencies of amino acids at residues minimally involved in MHC-I binding.

The model is trained using a dataset of 6,680 mutated peptides from various cancers with experimentally validated immunogenicity status, of which 596 are classified as immunogenic. The data was augmented by the addition of random natural peptides sampled from the human proteome, assumed to be non-immunogenic.

Albert et al. also used their antigen presentation predictor, *BigMHC-EL*, as the input for a predictor of immunogenicity, *BigMHC-IM*. This predictor was trained on non-random peptides in the *PRIME-1.0* and *PRIME-2.0* datasets. When compared to other models from the literature, *BigMHC-IM* was found to be the best-performing method for neoepitope prediction, whereas *PRIME-2.0* was found to be best for pathogen immunogenicity prediction [10].

Independent benchmarking of publicly available algorithms has suggested that current methodologies do not perform substantially better than random predictions, particularly for emerging viruses such as SARS-CoV-2, and for predicting cancer neoantigens [29]. The composition and volume of training data was believed to be a major source of the models' weaknesses, with skewed distributions of immunogenic/non-immunogenic peptides across different HLA types complicating prediction. In particular, no one method performed well in both pathogenic and neoantigen prediction contexts.

1.5 DPhil summary

The predictors of antigen processing parameters or processes reviewed in Section 1.3 are currently standalone predictors of the different steps shown in Figure 1.3. Although rudimentary efforts have been made to integrate proteasomal cleavage, TAP translocation and MHC-I binding affinity (e.g. *NetCTL*, which took a weighted sum of the 3 scores [93, 94]), no successful attempt has been made to combine the stages of antigen processing into a unified model of the pathway in order to predict raw pMHC abundance.

The primary objective of this DPhil is develop a mechanistic model of the antigen processing pathway, extending a systems biology peptide loading model presented by Dalchau et al. [43] to include Steps 1 to 5 in Figure 1.3. To do so, we develop our own predictive models of key parameters associated with these processes using common machine learning techniques, addressing key limitations discussed throughout Section 1.3.

In Chapter 2, we present a versatile framework for using proteasomal cleavage algorithm output to derive the probability of specific peptide products. We parametrise a simple probabilistic model based on this framework in order to be consistent with observed proteasomal product length distributions, resulting in a model which explains the scarcity of short (1-3 amino acid) products observed *in vitro*. We also consider direction of substrate entry and predict how this affects the probability of specific product formation. We conclude by comparing model predictions using three leading algorithms for cleavage prediction [115, 165, 175].

In Chapter 3, we develop a regressive model to predict peptide-TAP binding affinity: *PanTAP*. We expand training data availability by using datasets derived from mouse, rat and human TAP. This also enables us to form predictions across different species. By investigating different length normalisation strategies, we train and validate a model that can accommodate a range of substrate lengths.

Chapter 4 contributes a predictor of ERAP1 enzyme kinetics. A simple model is trained from publicly available data to accurately predict the results of a standard assay. The output of this is then converted to Michaelis Menten kinetics using a calibration curve, enabling direct integration into mechanistic modelling.

Chapter 5 sees the integration of the models in Chapters 3-4 with a systems biology approach inspired by Dalchau et al.'s peptide loading model [43]. We parametrise the model using experimental measurements of in an H2-Kb transfected cell line

and show how these parameters may be adjusted for other MHC-I alleles by considering the molecule's tapasin dependence. Chapter 6, sees further validation of the model's predictive efficacy using an experimental dataset of the processing efficiency of SIINFEKL precursors in H2-Kb transfected HeLa cells [71]. The use of an ERAP1 knockout cell line enables us to separately characterise the role of cytosolic aminopeptidases in the generation of the epitope.

The secondary objective of this thesis is to test whether the use of a mechanistic model of antigen processing results in an improvement in the prediction of CD8+ T-cell immunogenicity. We explore this in Chapter 7, training a novel classifier inspired by and trained using the data of Gfeller et al. from their *PRIME-2.0* tool [66]. We call the resulting predictor *POEM* (Prediction Of immunogenic Epitopes using Mechanistic modelling) and benchmark it against algorithms from the literature across multiple neoantigen and pathogenic datasets, finding performance that in some cases is significantly superior to the current state-of-the-art methods described in Section 1.4.

We conclude the thesis in Chapter 8 by summarising how the results of the previous chapters indicate that a mechanistic approach can yield comparable or superior predictive efficacy to machine learning approaches. We also propose how our other advantages of mechanistic modelling over machine learning may be leveraged by using our mechanistic model to simulate longitudinal studies of tumour evolution.

Chapter 2

Prediction of proteasomal cleavage products

In Chapter 1, we discussed and reviewed existing attempts to predict the probability of the amide bonds in a protein being cleaved by the constitutive or immunoproteasome [115, 165]. Although these algorithms display predictive efficacy in the classification of cleavage sites, they stop short of predicting the probability of specific peptides being produced from a source protein by proteasomal digest.

In this chapter, we develop and test new approaches to use the outputs of such models to predict the probability of formation of an epitope or its N-terminally extended precursors.

2.1 Introduction

The 26S proteasome is a protease complex residing in the cytosol that is responsible for the degradation of cellular proteins. The 26S proteasome complex is formed by the 20S core particle, in which the protein is cleaved, and two 19S regulatory particles that are responsible for substrate interaction. After recognition of a ubiquitinated protein substrate, the 19S regulatory particle unfolds the protein, removes the ubiquitin, and the substrate is fed through the barrel-like 20S core chamber, cleaving some, but not all, peptide bonds to produce short peptide products, typically between 3 and 22 amino acids in length (see Figure 2.1) [84]. It is through this process that the proteasome is believed to be predominantly responsible for the generation of the C-termini of MHC Class I epitopes [115]. The standard form of

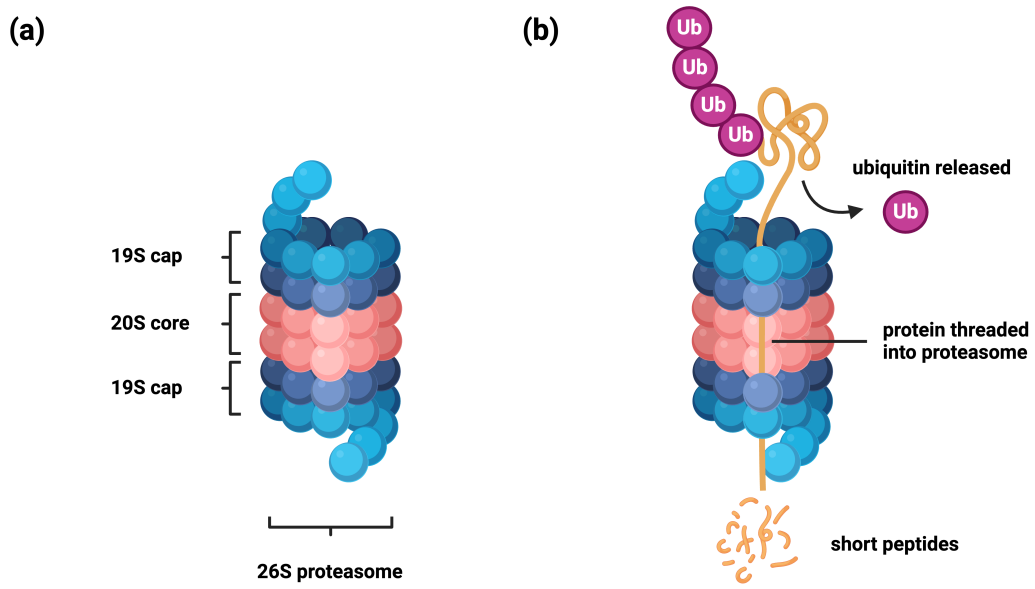


Figure 2.1. (a) Diagram of 26S proteasome subunits. (b) Diagram of ubiquitinated protein degradation by proteasome.

the proteasome found in mammalian cells is known as the constitutive proteasome. When stimulated by the cytokine interferon-γ (IFN-γ), three catalytic subunits in the 20S core are replaced, forming the immunoproteasome. This is characterised by a more efficient turnover and a slight change in cleavage specificity [57].

The direction of substrate entry to the proteasome is believed from MD simulations to depend on the energy required for the 19S cap to unfold either terminus [18] . This varies across substrates but has substantial consequences for the generation of epitopes. For example, ovalbumin has been found to preferentially enter the proteasome *in vivo* from its N-terminus [18]. Berko et al. blocked ovalbumin's N-terminus, inducing trimming from the C-terminus instead, which resulted in a large drop in the number of H2-Kb/SIINFEKL complexes on the cell surface. This demonstrates both the proteasome's important role in the generation of epitopes and the significance of the direction of substrate entry to the proteasome.

In order to predict epitope formation from source proteins, it is therefore desirable to be able to predict the probability of different bonds in the protein being cleaved as it passes through the proteasome. This task has received much attention in the literature since the publication of the Prediction Algorithm for Proteasomal Cleavages (PaProC) [89]. Simpler techniques were soon superseded by models trained using neural networks [115] and support vector machines [19]. In the past decade, newer algorithms have been published, benefiting from both larger training sets and advances in machine learning [165, 175]. Algorithms are trained on one of two kinds

of data: *in vitro* proteasomal digestion assays and large data sets of known Class I epitopes. Epitope data is often preferred because it is far more abundant than *in vitro* data, and is used to infer the activity of the proteasome in generating the C-terminus of the epitope.

These algorithms are all united by the fact that they do not explicitly predict the probability of a peptide being formed by proteasomal cleavage of the source protein. Instead, they return scores corresponding to the likelihood of each bond in the protein being cleaved. In their analysis, the authors of *NetChop* proposed a simple scaling factor to convert their predicted cleavage scores to probabilities. However, this was only an estimate to try to ensure approximate consistency with observed product length distributions, so further investigation is required to derive a strategy to convert the output of this and other algorithms into immunologically accurate yields of specific products.

Accordingly, in this chapter we develop and propose a probabilistic model for predicting specific peptide yield from the proteasomal digestion of a protein. We also consider the effect that the direction of protein entry to the proteasome has on predicted length distributions and specific product formation. Although this approach is flexible and may be applied to any cleavage score prediction algorithm, we choose focus on 3 families of algorithm from the literature: *NetChop* [115], *Pepsickle* [165] and a bidirectional long short-term memory model (*BiLSTM*) [175]. We fit our model for each of these algorithms using experimentally observed proteasomal product length distributions and test its ability to accurately predict specific product formation by using a study of ovalbumin digestion in constitutive and immunoproteasomes.

2.2 Methods

2.2.1 Probabilistic model formulation

In order to distinguish the probability of cleavage from the scores returned by different prediction algorithms (e.g. *NetChop*), we define the following notation:

$p_i :=$ probability bond at the C-terminus of amino acid i is cleaved,

$s_i :=$ algorithm cleavage score for bond at C-terminus of amino acid i,

where i denotes the index of the amino acid counting from the N-terminus of the peptide, starting with $i = 0$ (as in Figure 2.2).

2.2.1.1 Memoryless model

For the memoryless model, we assume the protein chain moves through the proteasome and each site is cleaved with a probability that is independent of cleavage events before or after it. To convert scores to probabilities, we introduce a scaling factor, γ. For a protein of length n, this yields the following cleavage probabilities

$$\begin{aligned} p_i &= \gamma \times s_i \qquad && 0 \leq i \leq n-2, \\ p_i &= 1 && i \in \{-1, n-1\}, \end{aligned} \tag{2.1}$$

where p_{-1}, p_{n-1} are not cleavage probabilities (as there are no bonds to cleave) but represent the fact that N- and C-termini of the protein are only bound to a single other amino acid.

Hence, the probability of a peptide spanning from residue i to residue $j > i$ being formed from a protein can be written by taking the product of the probabilities of successful and unsuccessful cleavage events:

$$P_{ij} = p_{i-1} \prod_{k=i}^{j-1} (1 - p_k) p_j. \tag{2.2}$$

Notably, this model results in the same probability of product formation whether the substrate enters the proteasome from its N- or C-terminus.

2.2.1.2 Non-memoryless model

Kisselev et al. observed very few products of lengths below 5 amino acids in their *in vitro* digestions [84]. This implies that cleavage sites near one another in the protein are somehow unlikely, which motivated the idea that cleavage may not occur independently at each bond in the protein.

Therefore, we propose an alternative model in which cleavage events occur with diminished probability at the residues directly following a successful cleavage event. Since the probability of a residue being cleaved in this model depends on the cleavage

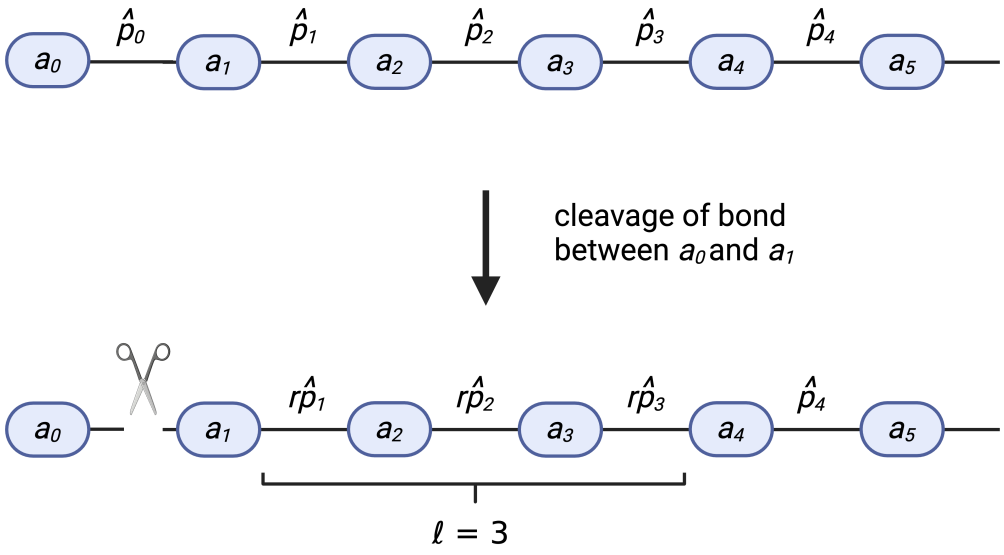


Figure 2.2. Schematic of non-memoryless model for the case $\ell = 3$ and assuming an N- to C-terminus trimming direction. Following successful cleavage of the first bond in the protein, the probability of the following 3 bonds being cleaved is scaled by the factor $r < 1$.

status of bonds before it, the system is no longer memoryless and we hence refer to this model as the non-memoryless model.

To capture this behaviour, we introduce two additional parameters: the number of residues for which cleavage activity is reduced, ℓ, and the scale of reduction in cleavage probability, r. This is shown in Figure 2.2 for the case $\ell = 3$. In the case $r = 1$, the non-memoryless model is equivalent to the memoryless model.

We assume that the output of the prediction algorithms is proportional to the total probability of a site being cleaved. By the Law of Total Probability, we can write this down in terms of these parameters as

$$\begin{aligned} p_i^N = &\hat{p}_i^N \times \mathbb{P}(\text{previous cleavage} > \ell \text{ aas towards N-terminus, or no previous cleavage}) \\ &+ r\hat{p}_i^N \times \mathbb{P}(\text{previous cleavage } \leq \ell \text{ aas towards N-terminus}) \end{aligned} \tag{2.3}$$

where $\hat{p}_i^N$ is the probability of a site being cleaved given that the previously cleavage was more than ℓ residues away towards the N-terminus. To simplify notation, we define the discrete random variable:

$$X_i := \text{Number of aas between previous cleavage site and site } i.$$

Equation 2.3 can therefore be written in terms of p_i^N, $\hat{p}_i^N$ and X_i as:

$$p_i^N = \hat{p}_i^N \cdot \mathbb{P}(X_i > \ell) + r\hat{p}_i^N \cdot \mathbb{P}(X_i \leq \ell), \tag{2.4}$$

for $0 \leq i \leq n-2$. We also impose that $p_{-1}^N = \hat{p}_{-1}^N = 1$ and $p_{n-1}^N = \hat{p}_{n-1}^N = 1$ to reflect the fact that the termini of the source protein are unbound.

Starting at the protein's N-terminus ($i = 0$), we define $X_0 = 1$. We can then write down expressions for the probability mass function (p.m.f.) of X_i for $0 < i \leq n-2$ by conditioning on the state of X_{i-1} and whether or not the previous bond was cleaved:

$$\begin{aligned}
&\mathbb{P}(X_i = 1) = p_{i-1}^N \\
&\mathbb{P}(X_i = k) = \mathbb{P}(X_{i-1} = k-1)\left(1 - r\hat{p}_{i-1}^N\right) && k \leq \ell + 1, \\
&\mathbb{P}(X_i = k) = \mathbb{P}(X_{i-1} = k-1)\left(1 - \hat{p}_{i-1}^N\right) && k > \ell + 1.
\end{aligned} \tag{2.5}$$

Hence, the p.m.f. of X_i can be defined in terms of the p.m.f. of X_{i-1}, p_{i-1}^N and $\hat{p}_{i-1}^N$. Consequently, Equation 2.4 can be solved for p_i^N if $\hat{p}_i^N$ is known, and vice versa.

We considered two possibilities for the incorporation of cleavage scores from prediction algorithms, s_i: (i) predicted scores are proportional to p_i, or (ii) predicted scores are proportional to $\hat{p}_i$.

Both possibilities were viewed as justifiable. The various prediction algorithms were trained to predict observed proteasomal cleavage sites, so are being trained to reproduce the total probability of cleavage, p_i^N. However, the input feature vector to all 3 algorithms is a frame of variable length, centered on the queried cleavage site. Hence, the algorithms are not forming their predictions from the entirety of the protein sequence, so could be argued to be predicting $\hat{p}_i^N$.

We tested both possibilities and found that, although both assumptions resulted in predictions that were largely consistent with the observed length distributions, the second assumption could more faithfully reproduce the data of Kisselev et al. for each of the three proteins. Hence, we converted scores to cleavage probabilities using:

$$\begin{aligned}
&\hat{p}_i^N = \gamma \times s_i && 0 \leq i \leq n-2, \\
&\hat{p}_i^N = 1 && i \in \{-1, n-1\},
\end{aligned} \tag{2.6}$$

for the non-memoryless model.

The probability of a peptide spanning from residue i to residue $j \geq i$ being formed from a protein can thus be written as:

$$\begin{aligned} P_{ij}^N &= p_{i-1}^N \prod_{k=i}^{k=i+\ell-1} (1 - r\hat{p}_k^N) \prod_{k=i+l}^{j-1} (1 - \hat{p}_k^N)\hat{p}_j^N \quad && j \geq i - 1 + \ell, \\ P_{ij}^N &= p_{i-1}^N \prod_{k=i}^{k=j-1} (1 - r\hat{p}_k^N)\hat{p}_j^N && j = i - 1 + \ell, \\ P_{ij}^N &= p_{i-1}^N \prod_{k=i}^{k=j-1} (1 - r\hat{p}_k^N)r\hat{p}_j^N && j < i - 1 + \ell. \end{aligned} \tag{2.7}$$

for a protein entering the proteasome from its N-terminus first.

2.2.1.3 Direction of protein entry

It has been observed that substrates may enter the proteasome from their C- or N-termini [18]. As mentioned, the memoryless model is symmetric in direction of entry. However, the non-memoryless model produces slightly different results depending on the direction of entry. Keeping the definition of X_i as the number of amino acids between site i and the previous cleavage site (now in the direction of the C-terminus), we modify Equations 2.5 for $0 \leq i < n - 2$, imposing $X_{n-2} = 1$:

$$\begin{aligned} &\mathbb{P}(X_i = 1) = p_{i+1}^C \\ &\mathbb{P}(X_i = k) = \mathbb{P}(X_{i+1} = k - 1)\left(1 - r\hat{p}_{i+1}^C\right) && k \leq \ell + 1, \\ &\mathbb{P}(X_i = k) = \mathbb{P}(X_{i+1} = k - 1)\left(1 - \hat{p}_{i+1}^C\right) && k > \ell + 1. \end{aligned} \tag{2.8}$$

We can therefore calculate p_i from $\hat{p}_i$ $(= \gamma \times s_i)$ using:

$$p_i^C = \hat{p}_i^C \cdot \mathbb{P}(X_i > \ell) + r\hat{p}_i^C \cdot \mathbb{P}(X_i \leq \ell), \tag{2.9}$$

The probability of a peptide spanning from residue i to residue $j \geq i$ being formed by a protein entering the proteasome from its C-terminus can therefore be written:

$$\begin{aligned} P_{ij}^C &= \hat{p}_{i-1}^C \prod_{k=i}^{k=j-\ell+1} (1-\hat{p}_k^C) \prod_{k=j-\ell}^{j-1} (1-r\hat{p}_k^C) p_j^C && j \geq i-1+\ell, \\ P_{ij}^C &= \hat{p}_{i-1}^C \prod_{k=i}^{j-1} (1-r\hat{p}_k^C) p_j^C && j = i-1+\ell, \\ P_{ij}^C &= r\hat{p}_{i-1}^C \prod_{k=i}^{j-1} (1-r\hat{p}_k^C) p_j^C && j < i-1+\ell. \end{aligned} \tag{2.10}$$

2.2.1.4 Prediction of direction of entry

It has been shown using molecular dynamics that the direction of entry correlates with the energy needed to unfold a protein from either end, with the lowest energy cost being favoured [18]. This theoretically provides a means by which the preferred direction of proteasomal entry could be predicted using molecular dynamics simulations. However, these simulations would be prohibitively computationally expensive for screening a large number of proteins. Hence, we conduct our analysis and model development for both directions and take the arithmetic mean probability of product formation, assuming a 50:50 split in entry direction for our final model. This was chosen as a parsimonious strategy given that no further information about direction of entry could viably be established.

2.2.2 Model fitting

2.2.2.1 Parametrisation dataset

The memoryless model has a single scale factor, γ, to estimate, whilst the non-memoryless model additionally has r and ℓ to determine. In order to estimate these parameters and select between different candidate models, we stipulate that our model should accurately replicate the length distribution of proteasomal products reported by Kisselev et al. [84]. The authors studied the *in vitro* digestion of ovalbumin, bovine casein and human insulin-like growth factor 1 (IGF) by mammalian 20s and 26s proteasomes. They report the length distributions of the products of the digestion of each protein. It should be noted that these proteins were denatured

before the experiment, so would not be expected to require further unfolding to enter the proteasome. Hence, this would suggest that all substrates were equally likely to enter the proteasome from either terminus.

We extracted these data using the Web Plot Digitizer [132] and retrieved protein sequences from UniProt for ovalbumin (P01012), casein (P02666) and IGF (P05019) in FASTA (amino acid sequence) format.

2.2.2.2 Cleavage prediction algorithms

Each FASTA file was initially passed through one of 6 different proteasomal cleavage prediction algorithms, returning scores, s_i, corresponding to the probability of each residue, i, in the protein being cleaved.

These algorithms are all trained using one of two types of data:

1. Data from *in vitro* digestion of proteins by proteasomes.
2. Data sets of known epitopes (e.g. IEDB).

In the second case, it is assumed that the proteasome generated the C-terminus of the epitope, giving *positive* examples of cleavage sites, whereas the proteasome did not cleave the remaining peptide bonds in the epitope, giving *negative* examples of cleavage sites. In this way, this binary labelled data can be used to train a predictive model of proteasomal cleavage. It should be noted that epitope data suffers from biases caused by peptide-MHC binding affinity and TAP translocation specificity, amongst other confounding variables. However, it is much more abundant than *in vitro* proteasomal digest assays, so is sometimes preferred for the training of proteasomal prediction algorithms.

Furthermore, the *in vitro* proteasomal digest assays may not be indicative of proteasomal behaviour *in vivo*. Depending on the timepoint chosen by the investigator, proteasomal products may have undergone subsequent digestion, giving an overrepresentation of certain cleavage events. The denaturing of substrates and the choice of substrate length may also create different conditions to those *in vivo*, significantly affecting the observed proteasomal activity. Other factors, such as the source of the proteasomes, the inclusion of a no-proteasome control, and the buffer conditions would also be expected to affect the data and may vary across studies. Hence, the *in vitro* proteasomal data may not be more suitable for training a predictive model than the epitope data.

Algorithm (version)	Immuno- / constitutive?	Training data type	Reference
NetChop	No	epitopes	[115]
NetChop (20s)	No	*in vitro* assays	[115]
Pepsickle (epitope)	No	epitopes	[165]
Pepsickle (in-vitro)	Yes	*in vitro* assays	[165]
Pepsickle (in-vitro-2)	Yes	*in vitro* assays	[165]
BiLSTM	No	epitope	[175]

Table 2.1. Summary of proteasomal cleavage prediction algorithms chosen for study in this thesis.

The algorithms considered are summarised in Table 2.1 and were chosen over other algorithms for investigation because of the superior efficacy they demonstrated in benchmarks of comparable proteasomal prediction methods [138, 165, 175].

For algorithms distinguishing between constitutive and immunoproteasome specificity we selected the constitutive version, since Kisselev et al. sourced their proteasomes from rabbit skeletal muscle due to their homogeneous composition of solely constitutive proteasomes.

The bidirectional long short-term memory (*BiLSTM*) predictor recently proposed by Ziegler et al., who reported state of the art performance, was not provided as a pre-trained model and had to be trained using the source code provided in the original paper [175]. We found comparable cross-validation performance to the reported figures, indicating that the training was a success.

2.2.2.3 Optimisation of model parameters

The resulting scores from each algorithm were passed to each of the two probabilistic models defined in Sections 2.2.1.1 and 2.2.1.2, from which the probability of each product of length 1 to 32 amino acids was calculated and used to determine the expected number of products of each length. This was preferred to a stochastic implementation in order to reduce noise in parametrisation and ensure reproducibility.

The predicted cumulative distributions were compared to Kisselev et al.'s empirical cumulative distributions and the total sum of squared errors (SSE) was taken as the

loss function. This loss function was minimised using the covariance matrix adaptive evolution strategy (CMA-ES) optimisation algorithm for the non-memoryless models, and the exponential natural evolution strategy (XNES) for the memoryless model. CMA-ES was selected as it has been reported to outperform other off the shelf optimisation algorithms on benchmark tasks. However, CMA-ES does not work for 1D optimisation, so the XNES algorithm was chosen for the memoryless model. Both algorithms were implemented by using the optimisation toolkit in the Probabilistic Inference on Noisy Time Series (PINTS) package in Python with default hyperparameters [40].

The parameter, ℓ, was fixed for each optimisation at a value in $\{1, 2, 3, 4, 5, 6\}$ because of its physical restriction to integer values (corresponding to number of residues). For $\ell \geq 6$, the sum of squared errors increased substantially, suggesting that that it was sufficient to restrict our search space to smaller natural numbers. The other parameters were permitted to take values $\gamma \in [0, 1]$ and $r \in [0, 1]$ and optimisations were initialised at random start points along the unit line or square respectively.

2.3 Results

2.3.1 Predictions of memoryless model

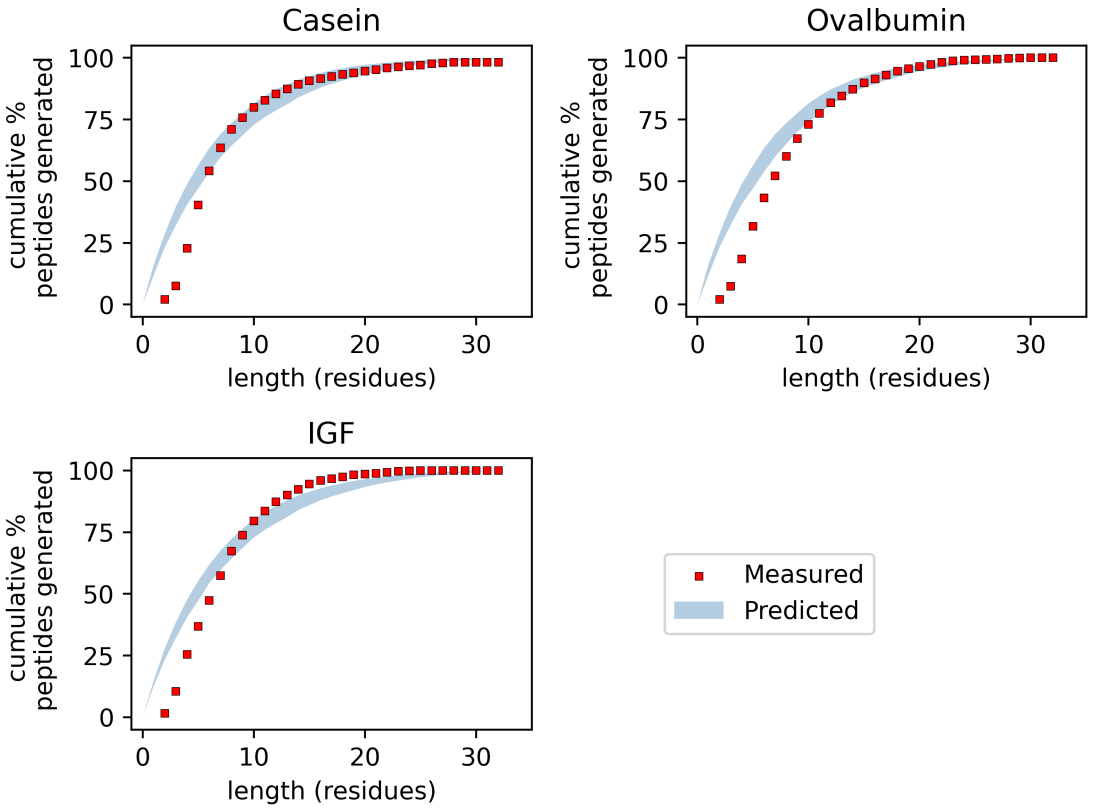


Figure 2.3. Illustrative plots for the 6 algorithms showing the range of model predictions for the memoryless model (blue shaded areas) compared to the observed Kisselev length distributions (shown by red squares).

The predictions of the memoryless model were found to consistently overestimate the number of products of short lengths (≤ 5 amino acids) across all 6 algorithms (see Figure 2.3). This motivated the development of the non-memoryless model, with a reduced rate of trimming following a successful cleavage in order to prevent short products from being formed.

2.3.2 Predictions of non-memoryless model

Motivated by the low number of products of short lengths, we altered the probabilistic model to incorporate reduced proteasomal activity for peptide bonds following a successful cleavage event. For all algorithms, we found that a reduction in activity for the following 3 peptide bonds after a successful hydrolysis was able to reproduce

the Kisselev length distributions far more faithfully than the memoryless model (see Figure 2.4).

The best performing of the 6 algorithms considered were the *BiLSTM* algorithm, which had the lowest mean squared error for casein, and the epitope version of the *NetChop* algorithm, which showed the highest efficacy for ovalbumin. The best performing algorithm on IGF was the *Pepsickle in-vitro* algorithm. However, this algortihm performed comparatively poorly on the prediction of casein and ovalbumin product lengths.

2.3.3 Effect of substrate entry direction on product length distribution

The direction of entry to the proteasome (C-terminus or N-terminus first) appeared to have very little effect on the distribution of substrate lengths predicted by the model. For every algorithm and protein, the lines corresponding to the distributions predicted for each direction are barely distinguishable in Figure 2.4. Differences in the overall quality of fit (indicated by MSE) did not reveal a large difference in efficacy for a specific direction for any of the proteins.

2.3.4 Prediction of ovalbumin digestion

To test and compare each algorithm's ability to accurately estimate the yield of specific products, we compared the predicted efficiency of generation of SIINFEKL and its N-terminally extended precursors to measured yields from Cascio et al.'s study of denatured ovalbumin digestion by 20s constitutive and immunoproteasomes [34]. As with the Kisselev data, we expected the denaturing of ovalbumin to result in the substrate being equally likely to enter the proteasome from each terminus, so plot the arithmetic mean of the predicted yield in Figure 2.5, with error bars showing the yield from either direction.

Where applicable, versions of models trained exclusively on constitutive or immunoproteasome data were used for the corresponding predictions.

Although the direction of proteasome entry was found to have negligible effect on the product length distributions (in Figure 2.4), it was found to affect the predictions of specific products (shown by the error bars in Figure 2.5). The largest effect was seen for the *NetChop* algorithm trained on epitope data. On average, the yield of the

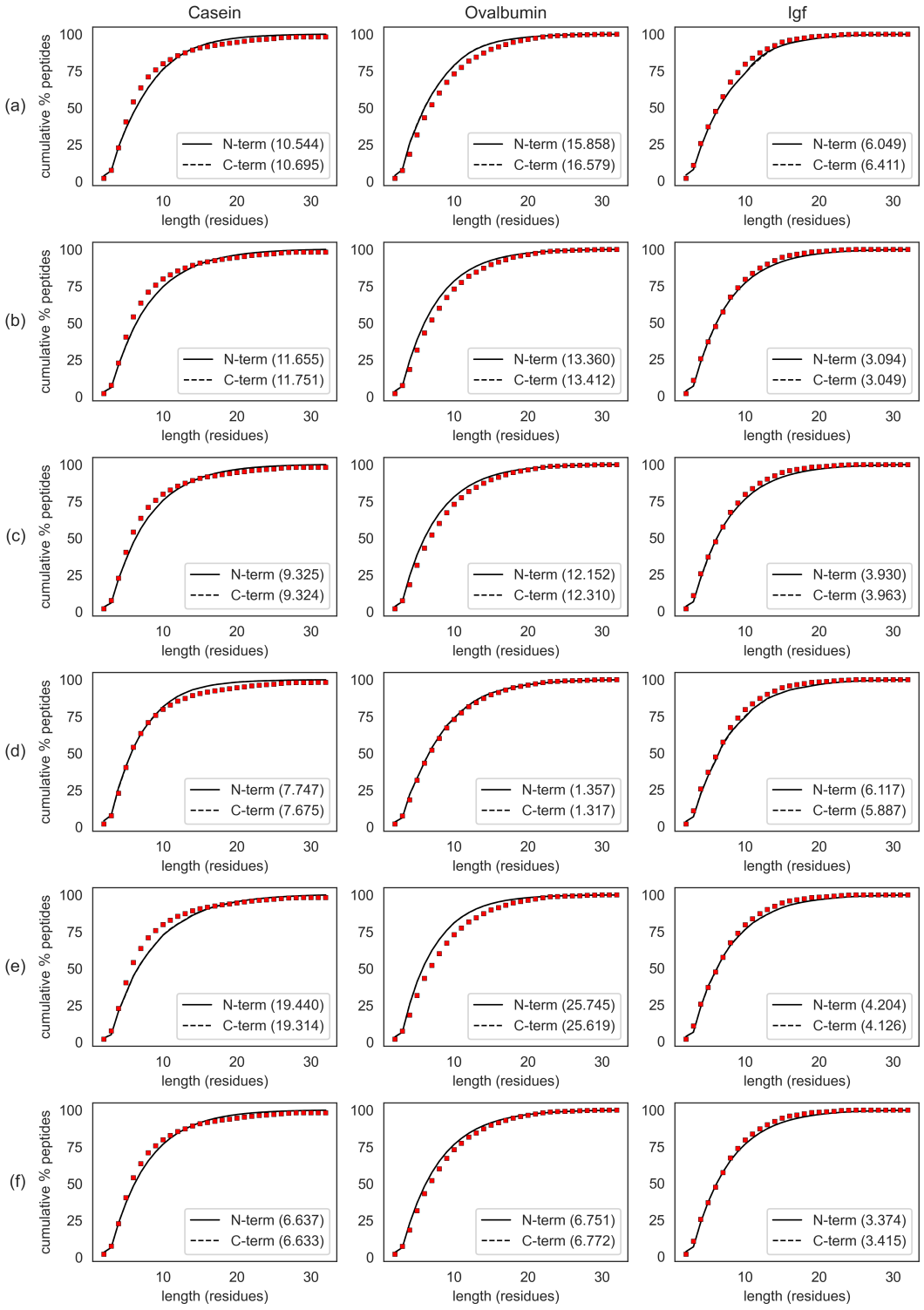


Figure 2.4. Comparison of algorithm performances on Kisselev dataset (shown in red), assuming protein entry from C- (dashed) or N-terminus (solid). Algorithms are: (a) *Pepsickle* (epitope), (b) *Pepsickle* (in-vitro), (c) *Pepsickle* (in-vitro-2), (d) *NetChop*, (e) *NetChop20s*, (f) *BiLSTM*. Mean squared error is shown for each method in the legend.

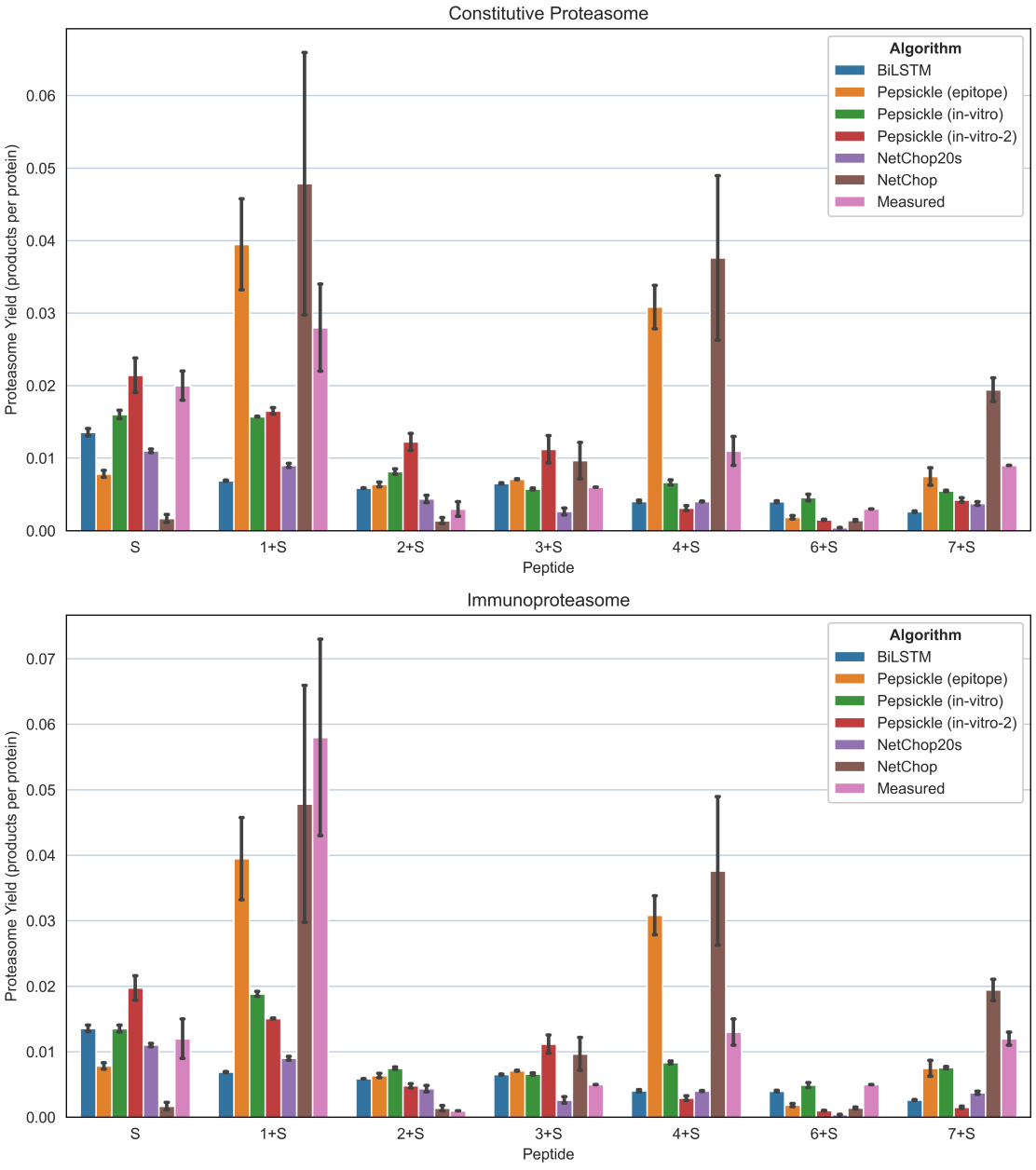


Figure 2.5. Comparison between predicted yields using different algorithms and measured yields of SIINFEKL (S) and its precursors (X + S) by Cascio et al. [34]. Error bars on predictions represent the predictions assuming entry from the N-terminus or the C-terminus. Error bars on the measured data represent reported standard errors.

Algorithm	Constitutive		Immunoproteasome	
	R_p	R_s	R_p	R_s
BiLSTM	0.515	0.541	0.110	0.200
Pepsickle (epitope)	0.714	0.955	0.825	0.927
Pepsickle (in-vitro)	0.887	0.649	0.882	0.818
Pepsickle (in-vitro-2)	0.643	0.541	0.445	0.273
NetChop	0.627	0.775	0.801	0.909
NetChop20S	0.859	0.685	0.553	0.418

Table 2.2. Correlation of predicted peptide yields with Cascio et al. measured yields following ovalbumin digestion by constitutive and immunoproteasomes. Correlation is given in terms of the Pearson correlation coefficient, R_p, and the Spearman correlation coefficient, R_s.

7 peptides was predicted by *NetChop* to be approximately 65% higher when entering from the C-terminus than from the N-terminus. This is due to the bonds following the phenylalanine and lysine residues near the C-terminus of SIINFEKL, which are given cleavage scores of 0.833 and 0.931 respectively by *NetChop*. Having entered the proteasome from the N-terminus, these bonds are highly likely to be cleaved. However, when entering from the C-terminus, these bonds are cleaved at a diminished rate following the successful cleavage of the leucine at the C-terminus. This results in a substantial increase in the production of SIINFEKL and its N-terminally extended precursors.

We found similarities in predictions across algorithms depending on whether they were trained on *in vitro* digestion assays or epitope data sets. Both the *Pepsickle* epitope version and *NetChop* predict high yields of 1+S and 4+S, whereas the *in vitro* versions predicted yields of approximately one third or less of this magnitude.

No algorithm correctly reproduced the hierarchy of product yields observed by Cascio et al. However, the maximum Pearson correlation coefficient across any model was $R_p = 0.887$ for the *Pepsickle* (in-vitro) model for the constitutive proteasome and $R_p = 0.882$ for the immunoproteasome. This was notably higher than any of the other 5 prediction algorithms across the two proteasomes, although *NetChop20S* showed similar efficacy on the constitutive proteasome study, and *Pepsickle* (epitope) and *NetChop* for the immunoproteasome data (see Table 2.2). Interestingly, the *BiLSTM* algorithm was the worst performing algorithm on both data sets, despite best fitting the product length distribution.

2.4 Discussion

2.4.1 Proteasomes inefficiently produce short peptides

Our analysis suggests that the assumption of independent cleavage by the proteasome across all sites in a substrate is incompatible with the low number of short (<3 aa) peptides produced by *in vitro* digests. This is not a unique feature of the Kisselev dataset — a scarcity of 2mers and 3mers was also observed by Nussbaum et al. following the digestion of enolase 1 [116].

It is possible that this could be an artefact of the size-exclusion chromatography (SEC) technique used to analyse proteasomal digest product lengths. Short products might be too small to be effectively separated by the SEC column, resulting in an underestimation of their yield. However, without strong evidence to support this, we assume that this phenomenon is the result of the structure and function of the proteasome.

To address this, we developed a model of proteasomal cleavage that includes a reduced rate of cleavage downstream of successfully cleaved bonds. We postulate that this reduced cleavage rate might be caused by the newly exposed terminus being unstably bound in the 20s subunit, becoming increasingly stable again as more of the polypeptide chain feeds through the proteasome. This is purely conjecture and would require further investigation. However, an investigation of proteasome structure is outside the scope of this modelling thesis and the underlying cause does not affect the results or application of our model.

2.4.2 The direction of substrate entry affects specific product formation but not length distribution

A study into ovalbumin cleavage upon entering the proteasome from either terminus found that activity was similar from either end *in vitro*, but that the specific products formed were different following digestion from the N- and C-termini [18]. This is another phenomenon that could not be explained by the memoryless model, since the predictions are independent of the direction of trimming.

In the non-memoryless model, whilst direction of entry was not predicted to affect length distributions, we predicted a small difference between the probabilities of product generation for all algorithms when trimmed from opposite termini, as shown by the error bars in the bar plots in Figure 2.5. Somewhat surprisingly, almost every

algorithm predicted an enhanced rate of SIINFEKL and precursor production when ovalbumin was trimmed from its C-terminus, despite Berko et al.'s observation that the N-terminus is preferred the preferred direction *in vivo*.

2.4.3 Choice of cleavage prediction algorithm

Our research did not yield an obvious choice of 'best' algorithm from the six considered. Although *BiLSTM* performed best in the prediction of product length distributions, it was the worst performing method for the prediction of specific products of ovalbumin digestion. This suggests that the performance of each algorithm in reproducing the Kisselev length distributions is the result of the distribution of cleavage scores returned by the algorithm rather than any indication of accurate prediction of specific cleavage site probabilities.

Although the ovalbumin yield data of Cascio et al. provided a helpful sanity check that our models are generating product probabilities of the correct magnitude, we found considerable incongruence between the observed product yields and most of the models' predictions. The best performing methods were the *Pepsickle* (in-vitro) and (epitope) methods, which were substantially better than all other methods across both studies. Cascio et al.'s data set was not included in the training set of the *Pepsickle* models, suggesting that this performance might be a fair representation of these models' predictive performance on unseen data.

Algorithms trained using epitope data (excluding *BiLSTM*) appeared to overestimate the production of 1+S and 4+S. This could be indicative of these models learning the rules of other features of the antigen processing pathway as a consequence of their training data. Interestingly, both of these peptides contain glutamic acid at the N-terminus. One interpretation could be that they are learning that glutamic acid is extremely inefficiently cleaved by ERAP1 and cytosolic aminopeptidases [6, 72], so is more likely to appear on the cell surface on the N-terminus of an epitope (ignoring the effect of MHC-I affinities).

This highlights the problem with using methods trained on such data, particularly for inclusion in a model of the antigen processing pathway in which some of these other features are going to be explicitly modelled. Utilisation of these methods puts one at risk of double-counting the effects of TAP translocation, cytosolic aminopeptidase specificity, ERAP1 specificity, MHC-I binding affinity, and other stages in the pathway.

2.4.4 Limitations

Our analysis in this chapter is heavily limited by the availability of appropriate data with which to test our models. We used only 3 proteins in a single study to fully parametrise our models to be consistent with observed length distributions. We then compared leading proteasomal prediction algorithms using a single study's reports of the production efficiency of only 7 different peptides, all from one of the proteins (ovalbumin) used during parametrisation.

We must therefore be very careful about the conclusions we draw from our results. Using such a small test set, it would be unwise to dismiss any of the prediction algorithms for lack of efficacy, although the performance of the *BiLSTM* algorithm should certainly be viewed as concerning.

2.4.5 Future work

In future, it may be possible to use the mechanistic model of the antigen processing pathway presented later in this thesis to train a predictor of proteasomal cleavage probabilities by using epitope data without the risk of biases introduced by the effects of by TAP, ERAP1 or MHC-I binding affinity. One could fix all parameters in the model except for the proteasomal component. The mechanistic model would then act as a head on the predictions of a neural network, for instance, returning proteasomal cleavage scores. Training of the neural network would then involve the mechanistic model being simulated for each epitope at each iteration. A possible complication here would be the additional computation required to simulate the model so many times. If this proved problematic, a reasonable approximation to the mechanistic model could be used as a surrogate to still capture the effects of TAP, ERAP1 and MHC-I loading.

2.4.6 Concluding remarks

In this chapter, we have introduced a probabilistic model to estimate the probability of specific peptide product formation by the proteasome. In order for this model to be consistent with reported length distributions in the literature, the probability of cleavage of each bond in the protein must be dependent on whether the bonds upstream of it are cleaved. This introduces a dependence on the direction in which the protein enters the protein — either from its N-terminus or C-terminus —- which is consistent with observations in the literature [18]. We compared three families of proteasomal cleavage prediction algorithms' abilities to predict ovalbumin digestion

and product length distributions. The latter revealed no major difference between the algorithms, whilst on the former task the *Pepsickle* in-vitro model showed the greatest efficacy.

In subsequent chapters, we show how this model integrates into a mechanistic model of the antigen processing pathway by providing a source of epitope and N-terminally extended precursors to the cytosol.

Chapter 3

Pan-species prediction of TAP binding affinity

After proteasomal digestion, epitopes and N-terminally extended epitope precursors are released into the cytosol. The cytosol is a harsh environment with many different amino-, carboxy- and endopeptidases rapidly destroying free peptides [166]. As a consequence, peptide half-life in the cytosol is generally only a matter of seconds [96]. For a peptide to be efficiently supplied to the endoplasmic reticulum (ER), it is therefore crucial that it is translocated from the cytosol to the ER before it is destroyed.

Peptides are translocated by the transporter associated with antigen processing (TAP) — a member of the ATP binding casette (ABC) family of transporters. In this chapter, we predict the efficiency of translocation of peptides by TAP by predicting binding affinities from peptide sequence. In doing do, we address limitations in existing approaches by the construction and use of a larger training set, and adopt a versatile framework that results in accurate estimation for peptides of epitope and precursor length.

3.1 Introduction

The transporter associated with antigen processing (TAP) is a heterodimer complex of TAP1 and TAP2 located in the membrane of the endoplasmic reticulum (ER). The current working model of the translocation mechanism of TAP is shown in Figure 3.1 and can be broken down into 4 stages:

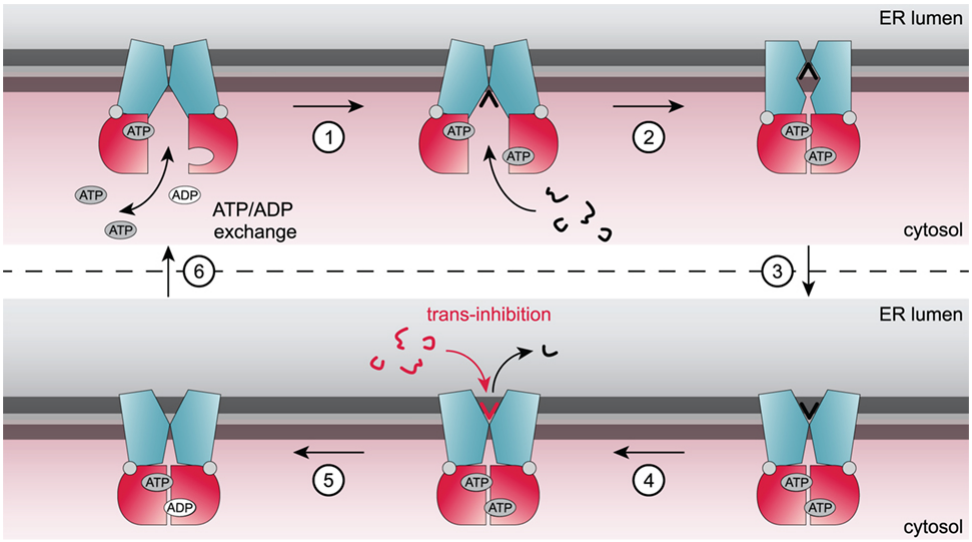


Figure 3.1. Schematic of the mechanism of action of peptide translocation by TAP. Figure taken from [100].

1. In the initial inward-facing state, the trans-membrane domains (TMDs) of TAP1 and TAP2 block peptides from diffusing into the ER lumen.

2. A peptide in the cytosol binds to the two TMDs. Independently of this, ATP binds to each of the non-binding domains (NBDs).

3. The peptide binding to TAP triggers an allosteric change in the TAP structure, bringing the NBDs together and switching the TAP to the outward-facing state, releasing the peptide into the ER lumen.

4. ATP is hydrolysed to form ADP. This destabilises the NBD dimer and causes the transporter to reset to its inward-facing, resting state.

In modelling the process of TAP translocation, we assume that the binding of ATP is rapid and not rate-limiting. Instead, we focus our attention on predicting the affinity of peptide binding to the TMDs and assume that the mechanism can be represented by a classical enzyme-substrate system:

$$\varnothing \underset{d_P^C}{\overset{g_i^C}{\rightleftharpoons}} P_i^C + S \underset{u_S}{\overset{b_i}{\rightleftharpoons}} SP_i \xrightarrow{k_S} P_i^{ER} + S, \tag{3.1}$$

where TAP is represented by S, the peptide by P_i and the superscripts ER and C are used to denote the compartment in which the peptide is located. In this chapter, we focus our attention on the prediction of the peptide-TAP binding affinity. Binding affinity is generally given in terms of the dissociation constant, $K_D := u_S/b_i$.

Prediction of peptide-TAP binding affinity is a well-studied problem in the literature, with continued attention from researchers over the past 20 years. Whilst advances in machine learning understanding and computational power have resulted in incremental improvements in prediction accuracy, existing algorithms for binding affinity prediction all share the same 3 limitations:

1. Lack of length diversity in training data (usually only 9mers) and/or lack of ability to form predictions for non nonameric peptides.
2. Training data and predictions only valid for humans.
3. Limited size of training data (generally the same 613 peptides used).

In this chapter, we aim to address all 3 of these problems through the creation of the first pan-species TAP binding affinity prediction model. This was in part motivated by the wealth of data available for measured TAP binding affinities using murine or rat TAP alleles, currently being unused for training predictive models [31, 69]. We were also inspired by the performance benefits gained by training a single model to predict peptide-MHC binding affinities across all MHC-I alleles in the development of *NetMHCpan* [114].

3.2 Methods

3.2.1 Construction of a training set

3.2.1.1 Data sources

The basis of our training set was the commonly used DS613 set of measured affinities (IC_{50}) for 613 different 9mer peptides using human TAP compiled by Diez-Rivero et al. [48]. We expanded this dataset through the addition of 65 additional human TAP binding affinities, expanding the diversity of substrate lengths to 8-15 amino acids.

The murine TAP binding affinity dataset of Burgevin et al. [31] was also incorporated, along with Gubler et al.'s measurements for the rat *a* and *u* haplotypes [69].

These data were predominantly discovered via the MHCBN database [95]. Where only available in paper figures, data was extracted using the Web Plot Digitizer [132]. The resulting dataset contained 1,343 peptides with binding affinities spanning over 5 orders of magnitude. This is summarised in Table 3.1.

Description	Species	# peptides	Lengths	Reference
DS613	Human	613	9	[69]
Burgevin	Mouse	323	8-11	[31]
Chang	Human	7	9	[36]
Daniel	Human	16	10-14	[44]
Van Endert	Human	23	8-12	[156]
Fleischhauer	Human	6	9-10	[59]
Gubler	Rat^a, Rat^u	342	9	[69]
Lucchiari-Hartz	Human	4	8-11	[103]
Uebel-95	Human	7	10-15	[154]
Uebel-97	Human	2	9	[155]

Table 3.1. Summary of curated pan-species TAP binding affinity data.

3.2.1.2 Standardisation of units

All discovered binding affinities were converted from their presented units into nM. Some data were discovered in terms of binding affinity relative to the K_D of the commonly used reference peptide, RRYNASTEL (R9L). We considered an IC_{50} of 400nM for R9L, in keeping with previous work [48]. Finally, the logarithm of base 10 was taken of all binding affinities to facilitate the model convergence and to avoid the prediction of any non-physical negative binding affinities.

3.2.1.3 Data preprocessing

A small subset of the dataset consisted of peptides which appeared 2 or 3 times. This was primarily due to peptides studied by Gubler et al. in their alanine scan assay with human, rat^a and rat^u TAP [69]. Repeated peptides from the same species were replaced by the geometric mean of their binding affinities in order to avoid putting excessive weight in the loss function on any one peptide. Repeated peptides from different species were kept separate in the hope that this would help the model to learn the rules behind inter-species differences in TAP binding.

3.2.2 TAP binding affinity predictive model training

3.2.2.1 TAP pseudosequence

A pseudosequence is a reduction of a full protein sequence to a specific subset of the residues. Adjacent amino acids in the pseudosequence are therefore not necessarily adjacent in the original protein sequence, but the order is typically preserved. To induce the regressive model to better learn inter-species differences in binding affinity, we reduced the TAP1/TAP2 reference sequences for each species to an 18 residue pseudosequence, as shown in Figure 3.2.

	TAP1					TAP2												
	296	304	412	459	467	213	217	218	262	265	266	270	373	374	380	399	425	429
Human	S	S	S	E	R	C	T	M	N	P	L	V	R	A	R	M	T	I
Mouse	C	S	V	Q	S	S	T	M	R	P	F	I	K	D	R	I	N	M
Rat a	C	N	V	E	S	S	A	E	Q	S	L	I	K	S	Q	I	N	M
Rat u	C	N	V	E	S	S	T	M	R	P	F	I	K	E	R	I	N	M

Figure 3.2. Pseudosequences for the 4 TAP alleles in our model.

To produce this sequence, we first reduced the primary structure to the 15 and 17 residues from TAP1 and TAP2 implicated in peptide binding by Lee et al.'s cryo-EM structures of TAP-bound 9mers [99]. 14 of these residues were identical across all species and alleles, so were dropped from the final pseudosequence as they would be expected to add no value to the prediction.

3.2.2.2 Regression technique selection

An initial comparison of regressors in the scikit learn Python package indicated that support vector regression (SVR) with a radial basis function (Gaussian) kernel was able to capture the non-linearities in the data better than the other candidate methods (SVR with linear kernel, lasso regression, and ridge regression).

SVR is generally viewed as better-suited to problems with small training datasets, so was preferred over a multi-layer perceptron. SVR is also a convex optimisation problem, so is guaranteed to converge to a unique solution. Furthermore, previous models of TAP binding affinity have also utilised SVR to good effect [20, 48], suggesting that the method is well-suited to this problem.

3.2.2.3 Peptide length standardisation

The peptides within our training set exhibit varying lengths, spanning from 8 to 15 amino acids. Given that regressive models require a consistent dimension of input features, it became imperative for us to standardise the length of the peptides in our training set. This standardisation was accomplished by the trimming of longer peptides and the padding of shorter ones.

To objectively evaluate the impact of different padding and trimming strategies, we systematically explored a spectrum of techniques. Each strategy can be uniquely represented by the target peptide length, k, and the padding or trimming start position, i. To illustrate, an n-mer is converted to length k by the following rules:

$$\mathrm{p}_1\mathrm{p}_2\ldots\mathrm{p}_n \rightarrow \begin{cases} \mathrm{p}_1\mathrm{p}_2\ldots\mathrm{p}_i\mathrm{p}_{n+i-k}\ldots\mathrm{p}_n & \text{if } n > k \\ \mathrm{p}_1\mathrm{p}_2\ldots\mathrm{p}_i\mathrm{X}_1\ldots\mathrm{X}_{k-n}\mathrm{p}_{i+1}\ldots\mathrm{p}_n & \text{if } n < k \\ \mathrm{p}_1\mathrm{p}_2\ldots\mathrm{p}_n & \text{if } n = k \end{cases}$$

where p_j denotes the amino acid at position j in the peptide and X is an unknown amino acid, often used as a padding token in such contexts.

The optimal padding and trimming strategy was selected through a preliminary experiment by encoding the peptide and amino acids with the BLOSUM62 substitution matrix and tuning hyperparameters as described in 3.2.2.6. The strategy returning the lowest mean squared error was used for the remainder of the analysis.

The final set of peptides was concatenated with the relevant pseudosequence, as outlined in Subsection 3.2.2.1.

3.2.2.4 Amino acid encoding

The amino acids in the concatenated peptide and pseudosequence were converted to numerical representations using three types of encoding.

1. A substitution matrix (e.g. BLOSUM62) using the BioPython package [37].
2. A matrix of physicochemical features from the AA Index database, reduced from 566 to 19 dimensions through principal component analysis, motivated by Wang et al.'s success in using these features to train neural networks [81, 163].

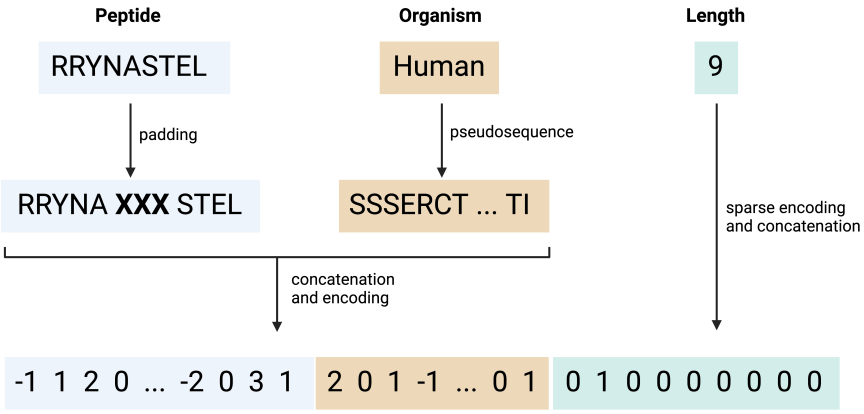


Figure 3.3. Summary of the process for producing input features for the regressive model. In this example, we are considering a padding strategy starting at position 5 and standardising to length 12.

3. A sparse encoding (sometimes referred to as one-hot encoding).

For each encoding, hyperparameters were tuned separately following the protocol in Subsection 3.2.2.6.

3.2.2.5 Length encoding

Although our input features must be of uniform length, we anticipated that the substrate length would affect the models predictions based on the observation that TAP binding affinity diminishes for substrates above 11 amino acids in length [73, 74, 141]. The original peptide length (i.e. before padding or trimming) was hence included in the input features using a sparse encoding vector of zeros of length 8, with a solitary 1 at the index corresponding to the peptide length minus 8. This sparse encoding was appended to the end of the peptide and pseudosequence encoding vector.

The construction of an input vector from the peptide sequence, TAP source organism and peptide length is summarised in the illustration in Figure 3.3.

3.2.2.6 Hyperparameter tuning

Hyperparameters were tuned using a randomised search across the 3-dimensional parameter space (as randomised searches are believed to be more efficient than grid searches [17]). Input features were scaled by either scikit-learn's *MinMaxScaler* (which normalises each feature between 0 and 1) or the *StandardScaler* (which fits a standard normal distribution to each feature) [125]. The SVR regularisation

Hyperparameter	Search distribution
Input scaling	{MinMaxScaler, StandardScaler}
Regularisation	$C \sim \text{Loguniform}(1, 100)$
Insensitive loss width	$\epsilon \sim \text{Loguniform}(0.001, 1)$

Table 3.2. Search space for randomised hyperparameter search.

parameter, C, was sampled from a log-uniform distribution between 1 and 100 and the width of the ϵ-insensitive loss was sampled from a log-uniform distribution between 0.001 and 1. These search space ranges were in part guided by the advice of Smets et al. [144].

For each run of the hyperparameter tuning, 250 sets of hyperparameters were randomly sampled and the resulting model performance evaluated by mean-squared error across a 10-fold cross-validation. The hyperparameter search domain are summarised in Table 3.2.

The randomised search and evaluation was carried out using scikit-learn's *RandomizedSearchCV* with a *Pipeline* to prevent any data leakage [125].

3.2.2.7 Cross-validation

As mentioned in subsection 3.2.2.6, model performance was evaluated by a 10-fold cross-validation. All peptides in Gubler et al.'s dataset occur in 3 copies in our training dataset — one for human TAP and another for each rat haplotype. We ensured that these peptides were separated into the same folds to prevent any inflated estimate of model generalisation by using scikit-learn's *GroupKFold* function. This guaranteed that any peptides in the test fold were completely unseen during model training.

3.2.2.8 Ensemble construction and selection

After hyperparameter tuning, we investigated whether taking an average ensemble of SVR models trained on different peptide encodings would result in an improvement in performance. We took all possible encoding combinations of size 2 to 10 and took the arithmetic mean of their predicted logarithmic binding affinities. The ensemble performance was evaluated using the same cross-validation splits as for the single encodings to enable a fair comparison.

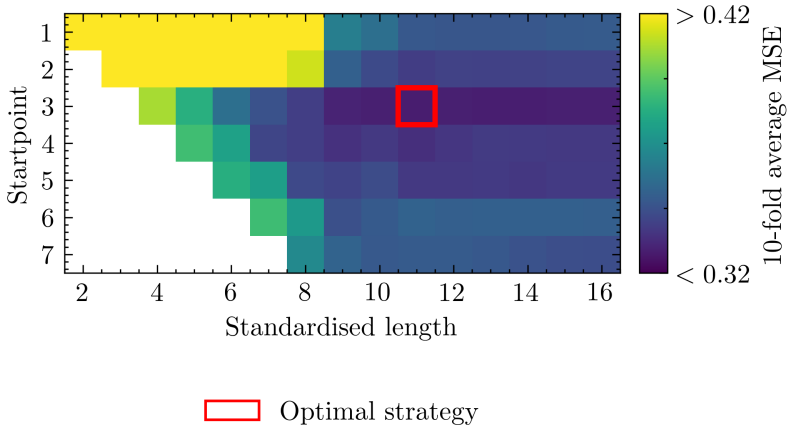


Figure 3.4. Comparison of different strategies to convert peptides to uniform length (colour corresponds to the log of the MSE to improve visibility).

3.3 Results

3.3.1 Effect of padding strategy

A systematic comparison of padding strategies revealed that starting padding or trimming after the first 3 amino acids from the N-terminus was the most effective strategy, as shown by the band of darker color in Figure 3.4. The best-performing standardised peptide length was 11 amino acids, although we did not find a significant difference between any lengths between 9 and 15 residues (using a paired one-tailed Wilcoxon test, $p > 0.05$).

3.3.2 Effect of amino acid encoding

The cross-validation performance of the best model for each encoding strategy is shown in Figure 3.5. We found that BLOSUM45 matrix was the best-performing encoding, resulting in significantly lower MSE than all encodings from Johnson to Dayhoff (paired one-tailed Wilcoxon test, $p < 0.05$). Matrices from the same family (i.e. BLOSUM and PAM) typically performed similarly to one another.

3.3.3 Results of ensemble construction

The effect of ensembling SVR models trained using different amino acid encodings is shown in Figure 3.6. We found that ensembling the BLOSUM45 and sparse models resulted in a substantial drop in MSE. Further additions to the ensemble contributed

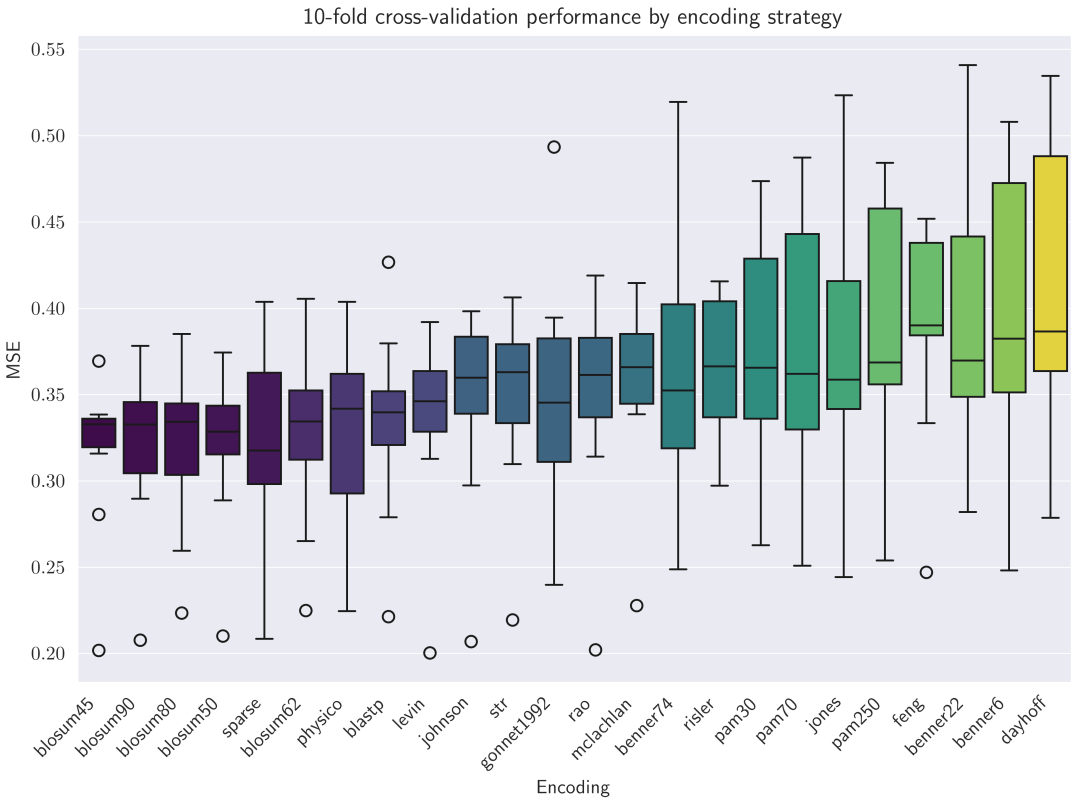


Figure 3.5. Accuracy of models trained using different amino acid encodings over 10-fold cross-validation. Methods are ordered by the average mean-squared-error across across all folds.

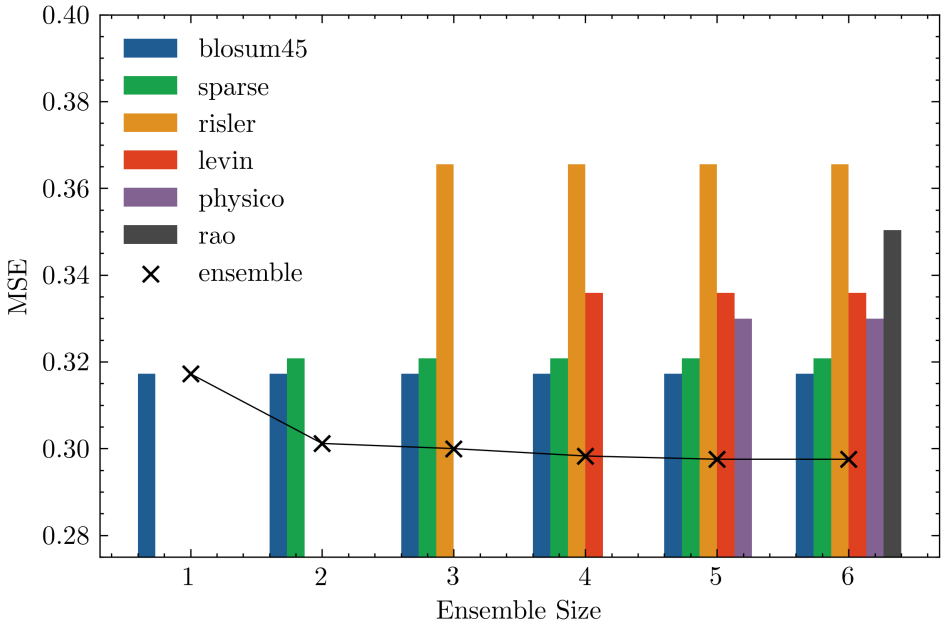


Figure 3.6. Effect of increasing ensemble size on MSE. Members of best ensemble of each size are shown by bars. MSE from average ensemble shown by superimposed line plot.

incremental improvements in overall model accuracy until an ensemble size of 6 was reached, beyond which any addition to the ensemble resulted in a loss of accuracy.

The members of the best performing ensemble method come from a diverse range of all three types of amino acid encoding strategy. No two versions of the same encoding (e.g. BLOSUM62 and BLOSUM45) were in our optimal ensemble.

Although the first two encodings to be added to our ensemble were amongst the top ranked methods in Figure 3.5, the risler and rao substitution matrices also ended up in our optimal ensemble, despite being in the bottom 50% of encodings on performance as individual models.

3.3.4 Performance by species

The 10-fold cross-validation predictions of the best performing ensemble were split by TAP allele and are plotted separately in Figure 3.7.

The best correlation between measured and predicted affinities was found for the *u* haplotype of rat TAP ($R_p = 0.9055$), which also had the lowest MSE. Conversely, the worst correlation was found for the other rat allele, Rat^a, due to many of the higher IC_{50} scores being underestimated by our model.

Both the human and murine datasets had substantially higher MSE than the rat datasets. We considered that this might be caused by the rat datasets solely consisting of 9mers, whereas the human and mouse datasets contain peptides of varying lengths. To test this idea, we re-calculated the MSE for only the 9mers in the human and mouse datasets. For the human data, this gave a modest drop in MSE from 0.3297 to 0.3239. For the mouse data, this led to an increase in MSE, from 0.3490 to 0.3736, thus disproving our hypothesis.

3.3.5 Performance by length

The effect of substrate length on predictive accuracy was evaluated using the 10-fold cross-validation predictions and is shown in Figure 3.8. The highest accuracy for well-represented lengths (making up over 2% of the total training data) was found for 9mers, which make up the majority of the training data (88.4%). Performance for lengths with few examples in the training set (12 to 15 residues) varied substantially.

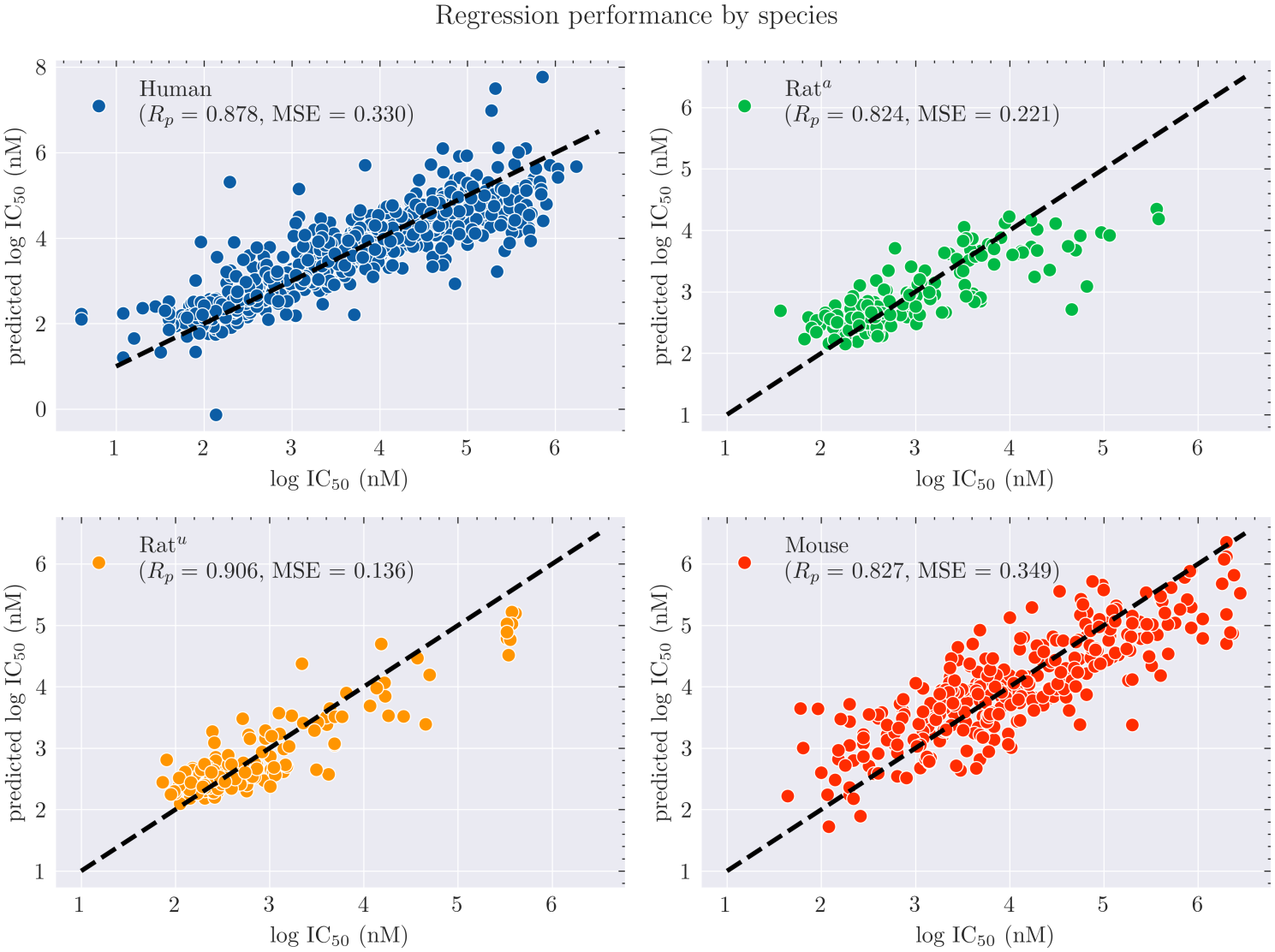


Figure 3.7. 10-fold cross-validation predictions plotted against true values for each species with line $y = x$ for reference. The Pearson correlation coefficient, R_p, for for each plot is included in the legend.

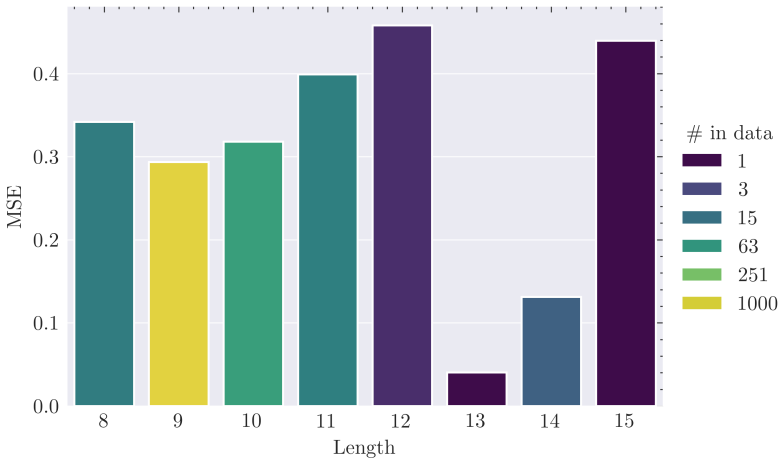


Figure 3.8. 10-fold cross-validation mean squared error by peptide length across all species. Bars are coloured by the number of peptides of that length in the training set.

Model	Pearson r	Spearman r	Reference
PanTAP	**0.8907**	**0.9077**	This thesis
DeepTAP	0.8906	0.9001	(Zhang et al.) [172]
TAPREG	-	0.89	(Diez-Rivero et al.) [48]
TAPPred	-	0.88	(Bhasin and Raghava) [20]
SVMTAP	-	0.82	(Donnes and Kohlbacher) [52]
ADM	-	0.74	(Doytchinova et al) [50]
SMM	-	0.82	(Peters et al.) [126]
PanTAP†	0.8901	0.8987	This thesis

Table 3.3. 10-fold cross-validation performance comparison on DS613 with other tools, reproduced from [172]. † model tuned and trained only on human binding affinity data.

3.3.6 DS613 benchmark against existing methods

3.3.6.1 Models in the literature

We include a comparison of performance on the standard DS613 dataset in Table 3.3. Our best ensemble model (which we henceforth refer to as *PanTAP*) displayed performance close to or better than the state-of-the-art, although caution should be exercised due to different cross-validation splits likely being used by the models. The performance of other models was taken from Zhang et al., who had collated performances from the original publications [172].

3.3.6.2 Human only model

To estimate the performance benefit or detriment *PanTAP* obtains from using pan-species data, we performed an ablation, repeating the entire model development process (from padding strategy selection through to ensemble selection) using only human data. This is included for comparison in Table 3.3. The pan-species data appears to have a small positive effect on both correlation coefficients, but well below the level required to be significant (tested through Fisher transformations and z-tests).

The effect is more pronounced for mean-squared error: *PanTAP* records an MSE of 0.3087 on the DS613 dataset, whereas the human only model had an MSE of 0.3135. However, a paired student t-test indicated that there was no significant difference between the MSEs of the two models ($p = 0.7153$, two-sided test).

3.4 Discussion

3.4.1 Optimal padding/trimming strategy aligns with anchor residues

The investigation of padding strategies revealed a clear preference towards the inclusion of the first 3 amino acids from the N-terminus. These, along with the C-terminus, are the sites identified by Uebel and Tampe as being key in the binding of peptides to TAP [154, 155]. What is perhaps surprising is that we did not find any benefit in also considering P4, since this has been recently reported by Lee et al. to interact with TAP1 in the binding of a 14mer [99]. However, Lee et al. only

present a structure for a single 14mer (LPAVVGLSPGEQEY), so this may not be indicative of how TAP binds to 14mers in general.

Although the lowest MSE came from a length of 11 amino acids, we found no significant difference between any lengths between 9 and 16 residues. This is consistent with what we would expect, since the central residues in structures of longer peptides bound to TAP are not in contact with either binding pocket [99], so their composition should not have a large impact on the binding affinity.

3.4.2 Pan-species prediction

One of the main motivations behind training a pan-species predictor was to be able to apply it to the modelling of antigen processing in pre-clinical models. *PanTAP* makes accurate predictions across 4 different TAP alleles and, through the passing of a pseudosequence, is theoretically able to make predictions for any sequenced TAP allele, so could be applied to the study of non-human primates, as well as the obvious application to mouse models. This could be useful in the elucidation of pre-clinical failures which may be down to the differences between animals and humans rather than limitations in the treatment being studied.

3.4.3 Ensembling of amino acid encodings

The ensembling of regressors trained using different representations of amino acids proved advantageous to performance. This mimics the findings of Jorgensen et al. who took an ensemble of models trained using sparse and BLOSUM62 peptide encodings and observed a similar improvement in performance during the training of *NetMHCstab* [80]. The fact that our final ensemble consisted of encodings with vastly differing performance as individual methods is consistent with the empirical observation that ensembles of diverse methods tend to yield bigger reductions in variance [88].

To our knowledge, this is the first systematic construction of an ensemble from the full range of substitution matrices available in the biopython package. However, the computation required to train and test so many models would be thoroughly impractical on a larger dataset. Since the performance of our size 2 ensemble of sparse and BLOSUM45 encodings was only slightly worse than the size 6 ensemble, and considering the success of the similar approach of Jorgensen et al., a reasonable strategy for a larger dataset would be to investigate a combination of only these two encodings.

3.4.4 Limitations

A major limitation of our model training is the lack of heterogeneity in peptide length in the training set. This makes it hard to estimate performance on non-nomameric peptides. Although model performance was found to be better for 9mers than other length peptides, this could be heavily driven by the fact that performance across the two rat datasets (both exclusively consisting of 9mers) was superior to that of the human and murine datasets.

Although the bulk of *in vitro* measurements for TAP binding affinity are for 9mers, it is believed that many of the translocated peptides *in vivo* will be N-terminally extended precursors of epitopes, so will likely be 10 amino acids in length, or longer. Hence, it is crucial that we have more confidence in our model's ability to predict binding affinities of longer substrates.

3.4.5 Comparison to previous work

To our knowledge, there is no published predictive model of TAP binding affinity designed to form predictions for multiple species. We anticipate that a pan-species model might be helpful tool in identifying discrepancies between human and animal antigen presentation, which may prevent erroneous conclusions being drawn from pre-clinical animal models.

Furthermore, most previous models were designed for prediction of 9mer binding affinity only, with the exception of Peters et al.'s stabilised matrix method [126]. This is a particular limitation of existing research because many translocated peptides *in vivo* are expected to be longer than the length required for loading to MHC-I, as can be inferred by ERAP1's effect on the peptidome [105]. In contrast, *DeepTAP* generates accurate predictions for peptides in the length range 8 to 15 residues (Figure 3.8).

3.4.6 Future directions

The combination of a larger training set from multiple species and the ensembling of different amino acid representations resulted in a model that was competitive with *DeepTAP*, which uses a sparse encoding and the DS613 dataset to train a recurrent neural network [172]. This demonstrates that the performance gain from using a deep learning approach may be outweighed by augmentations to the training set or increased focus on amino acid encodings.

Nonetheless, Zhang et al. were able to make incremental improvements on previous SVR-based methods by utilising RNNs, which would imply that the non-linearities in the TAP-peptide binding problem cannot be completely captured by a Gaussian kernel trick. Therefore, a logical next step would be to train a neural network using the PanTAP training set and an ensemble of sparse and BLOSUM45 encodings to see if performance can be improved upon any further.

3.4.7 Concluding remarks

To summarise, in this chapter we have presented the development of *PanTAP*: a novel predictor of TAP-peptide binding affinity. *PanTAP* displays competitive or superior efficacy on human 9mers to methods in the literature. Furthermore, *PanTAP*'s design means that it can be used to predict binding affinities for TAP alleles from other organisms, and for peptides of lengths between 8 and 16 amino acids. This flexibility in application should render it a valuable addition to the modelling of the antigen processing pipeline.

In Chapter 5, we shall incorporate *DeepTAP* into a mechanistic model of antigen processing and show how its predictions enhance the accuracy of this model.

Chapter 4

Prediction of epitope precursor N-terminus processing by ERAP1

4.1 Introduction

As a consequence of TAP's broad substrate length specificity and the lengths of proteasomal products [84], many of the peptides translocated to the endoplasmic reticulum are likely to still be of a greater length than is optimal for binding to MHC-I. These peptide are trimmed from their N-terminus by two aminopeptidases: ERAP1 and ERAP2.

For modelling of the antigen processing pathway, we wish to understand the role of N-terminally extended epitope precursors generated by the proteasome and how the removal of their extensions leads to the formation of the final epitope. ERAP1 and ERAP2 have also been observed to form heterodimers which appears to enhance the efficiency of the two enzymes and facilitate the trimming of longer peptides [137]. However, unlike ERAP1, the ERAP2 gene is not expressed in rodents or mice, and is not expressed as a full length protein in 25% of the human population. This points towards ERAP2 having only a minor, indirect involvement in the processing of epitope precursors. ERAP1's longer length preferences (8 to 16 amino acids) compared to ERAP2 (7 to 9 amino acids) would also suggest that ERAP1 is the enzyme that is primarily responsible for the removal of most epitopes' N-terminal extensions in the ER. Accordingly, in this chapter we focus on predicting peptide trimming by ERAP1.

ERAP1 has been noted to have a complicated substrate specificity, seemingly dependent on the amino acid composition at most sites in the peptide [55], as well

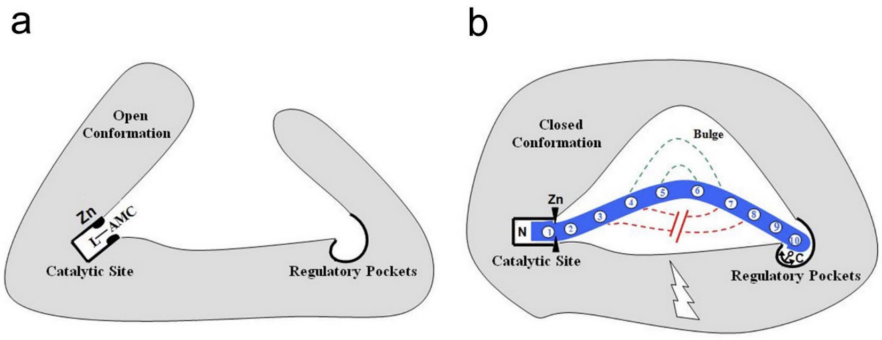


Figure 4.1. The molecular ruler mechanism: (a) A small substrate is unable to concurrently bind the catalytic site and regulatory pockets. ERAP1 therefore stays in the lower activity open conformation and hydrolysis of the substrate is inefficient. (b) A peptide of 9 to 10 amino acids is able to bind the regulatory domain from its C-terminus and reach the catalytic site from its N-terminus. This causes ERAP1 to undergo a conformational change into the higher-activity closed conformation. A single residue is then efficiently removed from the N-terminus. The central residues of longer peptides bulge into the internal cavity, as shown by the green path.

as the peptide length [35]. Studies of the structure of ERAP1 have provided an explanation for both of these dependencies through the proposed 'molecular ruler' mechanism (Figure 4.1). For a peptide to be efficiently trimmed, it must bind to the enzyme's regulatory pocket at its C-terminus, causing an allosteric change in ERAP1 conformation from the open to the closed state. The distance between the catalytic zinc site and the C-terminus has been reported as 29 Å [62]. Assuming an average contour length of 3.5 Å per amino acid [47], this creates a lower bound of 9 amino acids for a substrate to induce ERAP1 to enter the closed conformation. Accordingly, shorter substrates are trimmed with reduced efficiency. Despite not binding the active site of regulatory pocket, internal residues (i.e. not at the peptide N- or C-termini) also affect the strength of binding through interactions with the primary specificity (S1) pocket [171].

In this chapter we present a regressive predictive model to reproducing the results of an assay frequently used to measure ERAP1 activity in the literature. We then demonstrate how these predictions may be used to derive enzyme kinetic parameters for use in mechanistic modelling.

4.2 Methods

4.2.1 Construction of a training set

4.2.1.1 Type of training data

A thorough search of the relevant literature uncovered three different types of assay in which ERAP1 activity was measured and reported:

1. **Average trimming rate** – ERAP1 activity reported as the number of moles of substrate depleted over the course of a single time frame (often 1 hour).

2. **Enzyme kinetics** – reaction rate is measured as substrate concentration is increased. This is used to derive the catalytic rate, k_{cat}, and the Michaelis constant, K_M.

3. **Time series** – a fixed initial concentration of substrate is used. The remaining substrate is measured at different times. In some cases, the products are also tracked and their subsequent trimming by ERAP1 is measured [78].

Each type of data possesses unique advantages. Notably, assays that yield enzyme kinetics parameters offer the advantage of unambiguous extrapolation to contexts with varying enzyme or substrate concentrations, as well as predictable behavior in the presence of inhibitors. Importantly, these parameters can be seamlessly integrated into mechanistic models downstream, making assays resulting in the identification of enzyme kinetics parameters the most desirable for our specific research purposes.

However, after a thorough literature search, the first category of data proved to be substantially more abundant. To ensure the training of a predictive model with the most extensive and diverse dataset available, we opted to use these assays as the primary source of model training data.

We found 3 studies reporting this type of data for a large range of peptide lengths and amino acid compositions. In all cases, authors had used the same concentration of recombinant human ERAP1 (3.5 μg/ml) and initial substrate concentrations of between 100 and 150 μM. ERAP1 activity was reported in units of picomoles of substrate degraded per hour per nanogram of enzyme (pmol/h/ng).

We did not make adjustments for differences in initial substrate concentrations. This was based on the substantial excess of substrate relative to ERAP1 (assuming a molecular mass of 105 kDa), with a concentration ratio ranging from approximately 1:3000 in the 100μM case to 1:4500. Given this surplus of substrate compared to

the enzyme, differences in initial conditions were expected to have a negligible effect on the observed trimming rate.

4.2.1.2 Data inclusion criteria

Chang et al. measured the trimming rate of peptides longer than 16 amino acids. However, our focus is on training a model capable of predicting ERAP1 activity for peptides successfully translocated to the ER by TAP. It is believed that TAP primarily translocates peptides *in vivo* of lengths in the range 9-13 amino acids [131], although van Endert et al. showed similar binding affinity for sequences in the range 9-16 amino acids *in vitro* [157]. A significant drop in affinity was observed when peptides above this length were considered. When coupled with the low probability of the proteasome generating longer products, peptides of lengths above 16 amino acids are assumed to be a negligible minority in the ER.

Similarly, ERAP1 trimming of peptides below 9 amino acids in length has been found to be extremely inefficient *in vitro*, in accordance with the molecular ruler mechanism. However, Chang et al. report certain 8mers being trimmed at rates of approximately 5 pmol/h/ng. This level of trimming could have a significant effect on the presentation of 8mers through the class I presentation pathway, so we opted to include length 8 peptides in our training set. Shorter peptides were discarded as they were below the required length for presentation by MHC-I and are trimmed extremely inefficiently by ERAP1 [35].

York et al.'s paper only contained only a single peptide not found in Chang et al.'s paper (MIINFEKL). Rather than including the mean of the measurements in the two studies, we chose to include both data points for each peptide in our training set in the hope that the variance in training data might lead to a less biased estimator. Similarly, in Hearn et al's study of N-terminally extended SIINFEKL (S-L) peptides, the measurements are presented with error bars showing the individual outcomes of two different experiments. For some peptides, such as YS-L, this error bar is extremely large. We therefore decided to include both measurements for each peptide in this study.

We also initially tried to include the extensive 9mer trimming dataset of Evnouchidou et al., in which the authors systematically changed the amino acid at different sites in the peptide and measured ERAP1 activity [55]. However, the data is presented as the proportion of substrate degraded and the description of the protocol in the original publication raised uncertainties in the exact assay conditions used. Specifically, it was unclear whether the authors had incubated all solutions for the same length of

Description	# peptides	Lengths	Reference
Chang	70	8-16	[35]
York	26	8-14	[170]
Hearn	16	9	[72]

Table 4.1. Summary of curated ERAP1 trimming activity data.

time or whether they had used different lengths between 30 min and 4 hrs. Hence, this data could not be reliably included in our training set.

A breakdown of the data used for the final model training is shown in Table 4.1. The final training set consisted of 128 measurements corresponding to 85 unique peptides across 3 different studies.

4.2.2 Training

4.2.2.1 Selection of regression techniques

Our training set was only 10% of the size of the data set used to train *PanTAP* in Chapter 3, which permitted the testing of a wider range of candidate regressors.

After an initial comparison of regressors in the scikit-learn library, we again selected support vector regression (SVR) after finding superior efficacy compared to other techniques. However, to increase model heterogeneity in the final ensembles, we investigated the effect of using two different kernel functions, so included both linear and Gaussian/radial basis function (rbf) kernels in our hyperparameter tuning space.

4.2.3 Amino acid encoding and padding/trimming

We used the same combination of padding and trimming discussed in Chapter 3 to convert the peptides in the training set to uniform length. As our training set contained peptides up to 16 amino acids in length, we only tested padding strategies up to this length.

Whereas during the training of *PanTAP* we pre-screened padding and trimming strategies to cut down on computation, in this case the substantially smaller ERAP1 training set enabled us to tune hyperparameters and construct a separate ensemble for each padding/trimming strategy.

Amino acids were again encoded using one of three types of encoding:

1. A substitution matrix (e.g. BLOSUM62) from AA Index using the BioPython package [37, 81]
2. A matrix of physicochemical features from AA Index, reduced from 566 to 19 dimensions through principal component analysis, motivated by [163].
3. A sparse encoding (sometimes referred to as one-hot encoding)

4.2.3.1 Substrate length encoding

In contrast to the findings during the development of *PanTAP* in Chapter 3, we found that using a sparse encoding of the substrate length resulted in a drop in model accuracy, so chose not to include this in our regressor input features.

4.2.3.2 Hyperparameter tuning

For each combination of padding strategy and encoding, we tuned hyperparameters using 10-fold cross-validation. To avoid overestimating model generalisation, we ensured that all identical peptides were placed into the same splits by using the *GroupKFold* function in scikit-learn [125].

To ensure that the SVR model placed equal importance on every peptide sequence in its training set when determining its hyperplane, we weighted the input data inversely proportionally to the number of appearances of each peptide in the training data. Peptides appearing twice (e.g. from Chang and from York) were assigned a weight of 1/2 and ESIINFEKL (which appears 4 times in the data) was assigned a weight of 1/4.

We then tuned hyperparameters using the *RandomizedSearchCV* function in scikit-learn to sample the 4-dimensional parameter space shown in Table 4.2. For each run, 250 sets of hyperparameters were randomly sampled and the resulting model performance evaluated by mean-squared error across the 10 folds.

4.2.3.3 Ensemble generation

For each padding strategy and for each encoding, we took the sets of hyperparameters corresponding to each of best performing models. To introduce additional model heterogeneity into the ensemble, we took the best performing SVR model trained

Hyperparameter	Search distribution
Input scaling	{MinMaxScaler, StandardScaler}
Regularisation	$C \sim \text{Loguniform}(1, 100)$
Insensitive loss width	$\epsilon \sim \text{Loguniform}(0.001, 1)$
Kernel function	{rbf, linear}

Table 4.2. Search space for randomised hyperparameter search.

using a Gaussian kernel and the best with a linear kernel, yielding two models per encoding.

We then grouped these models across all encoding strategies and constructed ensembles of up to a maximum size of 5 encodings (choosing this initial upper limit to prevent the problem from becoming computationally prohibitive). The best ensemble was selected for each padding strategy by choosing the ensemble with the lowest mean squared error (MSE) across the 10-fold cross-validation.

4.2.4 Conversion to Michaelis-Menten kinetics

Our regression model was specifically crafted to forecast the precise outcome of a trimming assay, namely, the average rate at which an initial concentration of 100 to 150 μM of peptide is trimmed by ERAP1 *in vitro* in the absence of any competition. However, these reaction conditions are far from the expected conditions in the endoplasmic reticulum, in which we would expect significantly lower substrate concentrations [100] and competition from a multitude of substrates to bind to ERAP1. We therefore converted the predicted trimming efficiency in this assay to Michaelis-Menten kinetic parameters.

4.2.4.1 Estimating Michaelis-Menten kinetic parameters

To understand the relationship between Michaelis-Menten kinetics and the results of the trimming assay, we used a time-series of XS-L digestion over 180 minutes by ERAP1 to derive Michaelis-Menten kinetics for the N-terminally extended SIINFEKL peptides in the Hearn et al. dataset [72].

Michaelis-Menten parameter estimation suffers from issues with practical identifiability, particularly in the case of single enzyme and substrate initial concentrations [38]. To address this, we made the assumption that the Michaelis constant, K_M,

is not substrate-specific and fixed it at 100 μM for all substrates falling within the range of 8 to 16 residues in length (beyond which ERAP1 activity swiftly falls [35]). This value was chosen because it is a commonly used initial concentration in ERAP1 enzymatic assays.

To assess the identifiability of this parameter from the time series data, we used a Bayesian inference framework to generate the posterior distribution of each catalytic rate, k_X, (where X denotes the N-terminus amino acid) using Markov Chain Monte Carlo (MCMC) sampling. The rate of degradation of each substrate is described by the differential equation

$$\frac{\mathrm{d}P_X}{\mathrm{d}t} = -E_0 k_X \frac{P_X}{K_M + P_X}, \tag{4.1}$$

where E_0 denotes the initial concentration of ERAP1 and is fixed at 0.0325 μM.

We assumed a multiplicative Gaussian log-likelihood, under the assumption that the measurement error would likely be proportional to the magnitude of the fluorescence. For each sampled value of the catalytic rate, k_X, the log-likelihood of this parameter given the observed data, P_X^{obs}, is equal to

$$\ell\left(k_X | P_X^{obs}\right) = -\frac{n_t}{2}\log 2\pi - \sum_{i=1}^{n_t} \log P_X(t_i)\sigma - \frac{1}{2}\sum_{i=1}^{n_t}\left(\frac{P_X^{obs}(t_i) - P_X(t_i)}{P_X(t_i)\sigma}\right)^2, \tag{4.2}$$

where n_t denotes the number of time-points measured by Hearn et al. and t_i denotes the corresponding time at which the measurement was taken. The parameter σ determines the size of the standard deviation relative to the size of the observable. We considered values for σ between 0.02 and 0.1 and restricted it to be the same across all 16 amino acids measured (under the assumption that the experimental noise is the result of the apparatus and protocol used, rather than a property of the substrates).

Using the PINTS package in Python, we constructed 3 Markov chains using the Haario-Bardenet algorithm [40]. After 100,000 iterations, we ensured that all chains had converged by confirming that the convergence criterion, $\hat{r}$, was less than 1.05 for all sampled parameters (as recommended in the Stan documentation, which is often used as a guide for inference [33]).

We also derived a single set of maximum likelihood parameters from this log-likelihood function using the covariance matrix adaptation evolution strategy (CMA-ES) in the

optimisation toolkit in PINTS [40].

4.2.4.2 Calibrating MM kinetics and trimming rates

We used the maximum likelihood estimates of k_X for the 16 peptides in the Hearn dataset to construct a calibration curve between inferred Michaelis-Menten catalytic rates and the average degradation rate reported in the paper. We did this for two reasons:

1. The average degradation rates were presented as bar plots with error bars showing the outcomes of two individual experiments. It was unclear which data point corresponded to the time series data.

2. It was not apparent how the authors had calculated average degradation rate across the 16 peptides (i.e. which time point had been chosen as the cut-off to calculate the rate). Hence, we could not use an analytical approach to convert between the two.

The resulting calibration plots were used to identify which of the two experiments corresponded to the time series. The line $y = mx$ was fitted to this dataset and used to deduce the appropriate scaling factor for converting average degradation rates predicted by our regressive model to k_X.

4.3 Results

4.3.1 Effect of padding strategy on predictive performance

We compared different strategies for converting our training peptides to a uniform length across the data set. For each padding strategy, the best ensemble formed from the trained regressors using that strategy was taken and its accuracy measured using MSE over a 10-fold cross-validation. A comparison of the different padding strategies is shown in Figure 4.2.

We found a clear benefit to commencing padding or trimming after the 5th amino acid from the N-terminus, although padding after the 3rd amino acid was also seemingly better than other strategies, as shown by the dark colours on the rows corresponding to these two start points. The optimal length strategy was deemed to be standardising all peptides to 11 amino acids in length. Shorter lengths were found

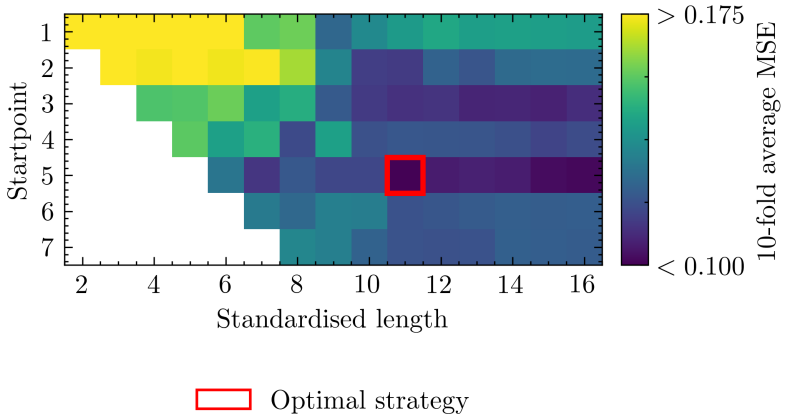


Figure 4.2. Comparison of best-performing ensembles for different padding strategies. Colour corresponds to 10-fold cross-validation MSE (log-scaled for clearer visualisation). The best performing method is highlighted.

to be significantly worse (paired Wilcoxon test with $p < 0.05$), whereas standardising peptides to longer lengths caused a non-significant drop in accuracy.

4.3.2 Effect of encoding on predictive performance

We compared the performance of regressors trained using one of 26 different amino acid encoding strategies and a choice of two kernel functions. In almost all cases, the Gaussian kernel ('rbf') resulted in a lower median MSE than using a linear kernel. The few exceptions to this were all amongst the worst performing encodings (e.g. sparse or benner6).

Regressors trained using the point accepted mutation (PAM) matrix family showed a notable difference between rbf and linear kernels, suggesting that this encoding introduces more complicated nonlinearities within the data than alternative encodings.

We did not find a significant difference between the predictive accuracy of most of the encodings. The best rbf regressor, using the McLachlan substitution matrix, was compared to the other regressors using a paired Wilcoxon test across the 10-fold cross validation performance. Only the sparse, risler, physico and rao regressors (using rbf kernels) were significantly less accurate at a 5% significance level.

4.3.3 Results of ensemble construction

We constructed ensembles up to a maximum size of 5 for each padding strategy from the best performing regressors using linear and Gaussian kernels. For every padding strategy, we found that an ensemble of regressors trained using different encodings

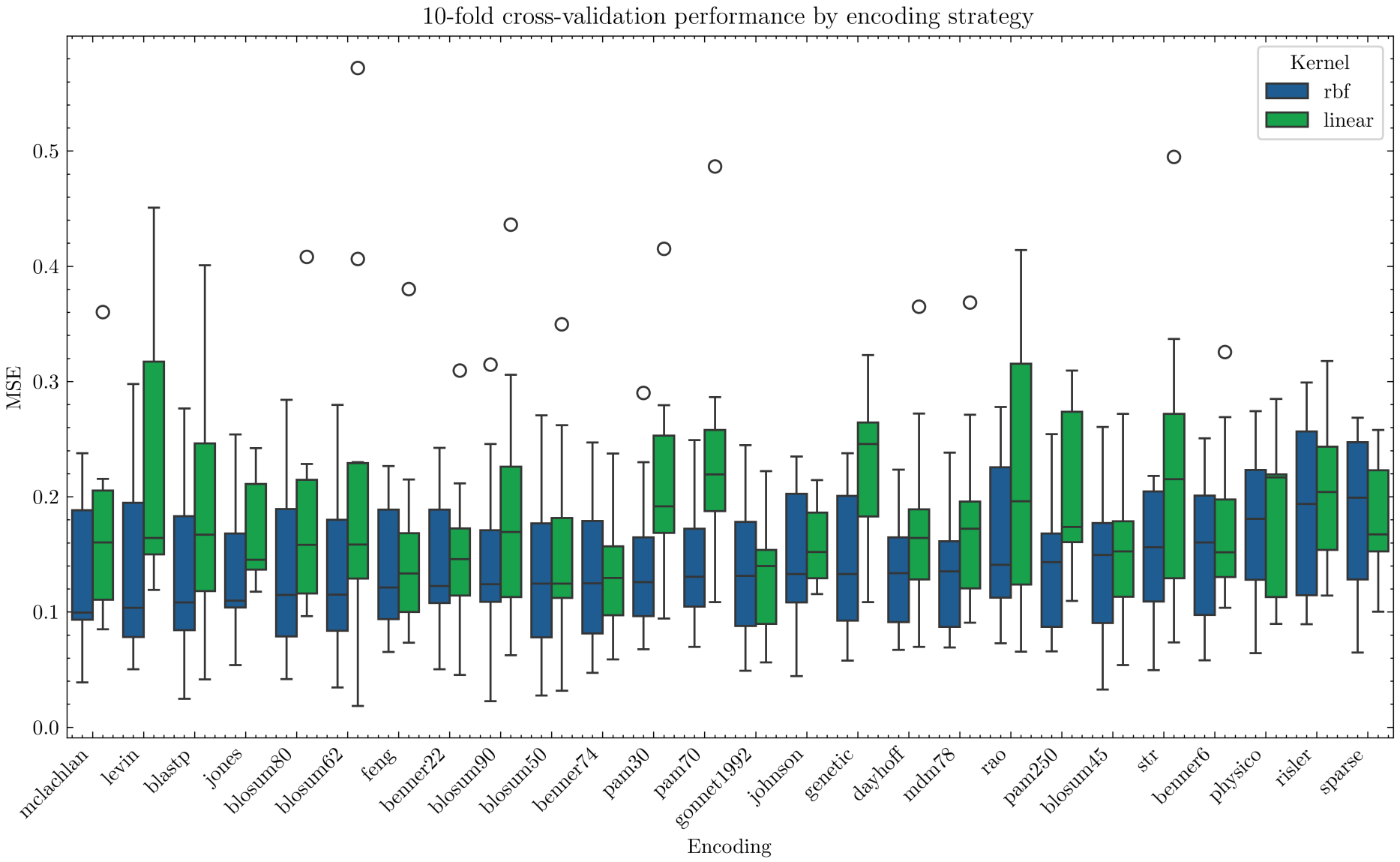


Figure 4.3. Boxplots showing accuracy through 10 fold cross-validation performance of the hyperparameter tuned regressors using Gaussian (green) and linear (blue) kernels with the optimal padding and trimming strategy.

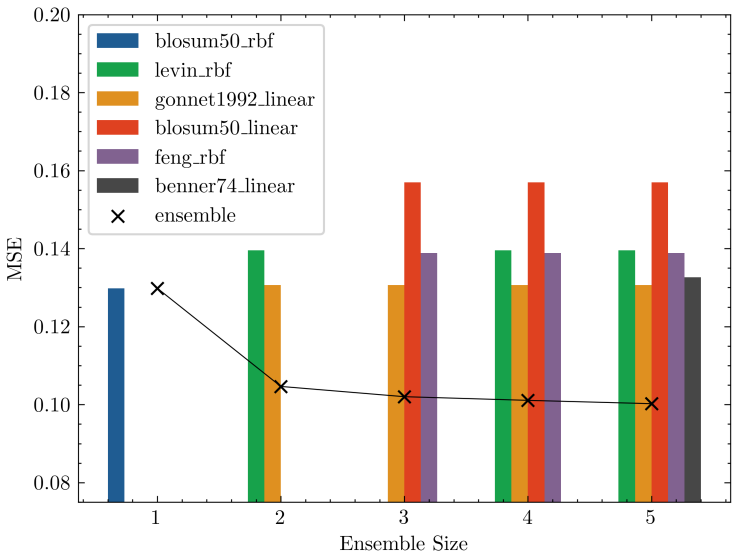


Figure 4.4. Effect of increasing ensemble size on MSE. Members of best ensemble of each size are shown by bars with height indicating the accuracy of the individual model, as determined by MSE across a 10-fold cross-validation. The performance of the ensemble of methods is shown by the superimposed line plot.

outperformed any individual model. The best ensembles for each size between 1 and 5 are shown in Figure 4.3 for the optimal padding strategy from Figure 4.2.

Although the BLOSUM50 encoding with a Gaussian kernel was the best performing regressor in isolation, this regressor did not appear in the best ensemble of any size from 2 to 5. Instead, each ensemble contains a combination of regressors using both linear and Gaussian kernels (despite the poorer individual performance of the linear kernels seen in Figure 4.3). The most accurate ensemble was the largest ensemble tested, containing 5 different encodings. Once again, the largest improvement in performance came in the jump from 1 to 2 models, just as we found in the TAP model development.

4.3.4 Final model performance

The predictions of the size 5 ensemble, made during a 10-fold cross-validation, are shown in Figure 4.5. We found strong correlations between predictions and the training data, with a mean squared error of 0.1002 and Pearson and Spearman correlation coefficients of 0.7947 and 0.7685 respectively. A drop in performance for peptides with low measured trimming rates can be seen from the loose packing of points in Figure 4.5.

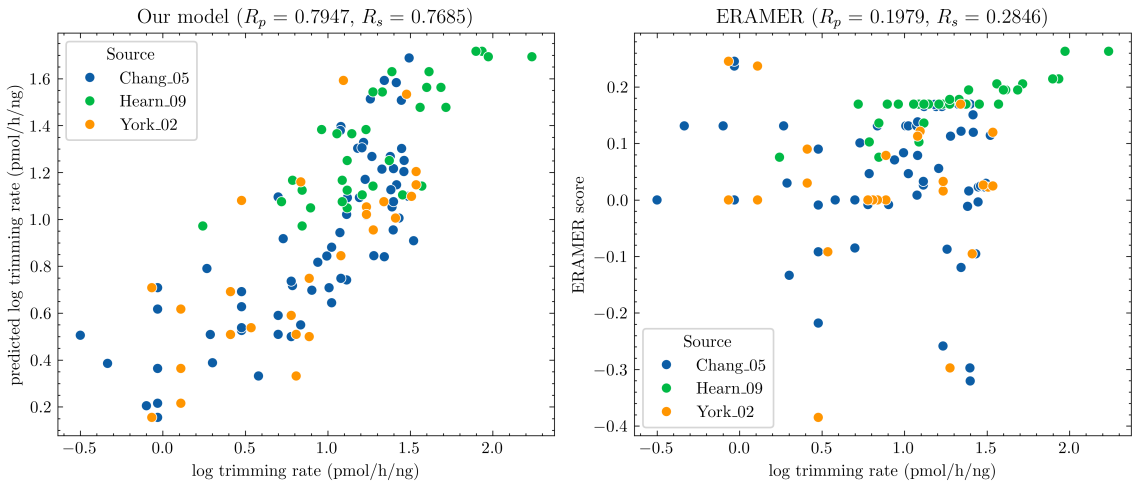


Figure 4.5. Comparison of 10-fold model predictions with predictions from ERAMER. Data points are coloured by the paper they were sourced from.

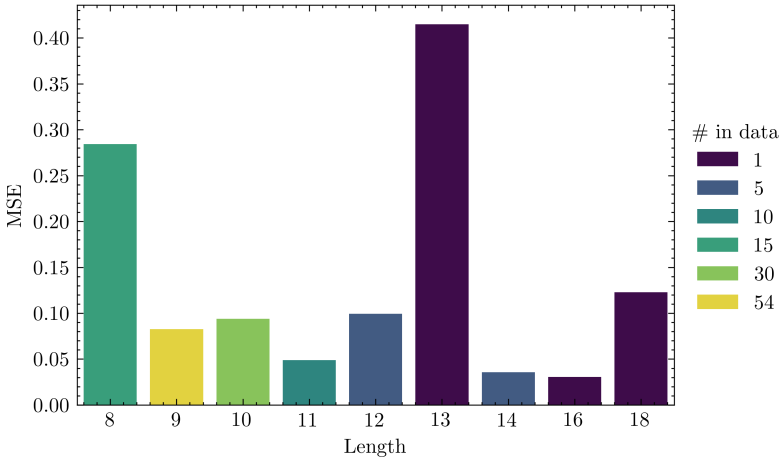


Figure 4.6. 10-fold mean squared error split by substrate length. Bar colour indicates the number of examples in our training set of the corresponding length.

4.3.4.1 Length bias

We theorised that the poor performance for inefficiently trimmed peptides might be related to a length bias in the model. To test this, we looked at predictive performance separated by peptide length. Strikingly, mean squared error for 8mers was 0.28, which was substantially higher than the mean across the training set, particularly for peptides in the range 9 to 12 residues in length (Figure 4.6). However, due to the low number of samples, a Mann-Whitney U test indicated that there was not a significant difference between the MSEs of the 8mers and those of the longer peptides ($p = 0.1273$).

4.3.4.2 Benchmarking against ERAMER

A systematic benchmarking against other methods in the literature was not possible because we were only able to find a single algorithm for ERAP1 prediction: *ERAMER* [8]. This method uses scoring matrices to return a score between -1 and 1 for any peptide based on its length and amino acids at different sites.

The algorithm was accessed using the provided Github repository on 14/1/24 and used to form predictions on the training set. The performance on our training set is shown for comparison in Figure 4.5. It should be noted that *ERAMER*'s scores may not be linearly proportional to the rate of trimming, so the Spearman coefficient is a fairer indication of the model's predictive efficacy than the Pearson coefficient.

ERAMER performs well on the Hearn XS-L peptides ($R_p = 0.7550$, $R_s = 0.8696$). However, it should be noted that this study was used by the authors to derive the matrix coefficients. Similarly, the model shows predictive capability for a subset of the Chang dataset corresponding to RYWANATRSX trimming by ERAP1 ($R_p = 0.5525$, $R_s = 0.7702$) but not for the remaining data in the paper ($R_p = 0.0655$, $R_s = 0.1764$). The authors mention using the Chang dataset to develop their model but this disparity in performance would suggest that they only used the C-terminus study and not the remainder of the data.

Performance on the York dataset is even worse ($R_p = 0.0066$, $R_s = 0.0789$). Overall, this suggests that *ERAMER* does not generalise well to unseen data and is overfitting to its training sets.

4.3.5 Michaelis-Menten kinetics for ERAP1

Figure 4.7 shows that Michaelis-Menten kinetics with K_M fixed at 100 μM are compatible with the *in vitro* digestion of the 16 SIINFEKL precursors studied by Hearn et al. The shaded regions denote the range between the 5th and 95th percentile of predicted substrate remaining at each time after a sample of 5,000 parameters was drawn from the posterior distributions of each catalytic rate, k_X. These show a tight spread across the posterior, indicating high certainty in the inferred parameters.

4.3.6 Calibration curve for converting scores to catalytic rates

We fitted a simple linear regression with no bias term (to ensure that the resulting line passed through the origin) between the maximum likelihood catalytic rates from 4.3.6 and the mean trimming rates from the two repeats, as shown in Figure 4.8. We found

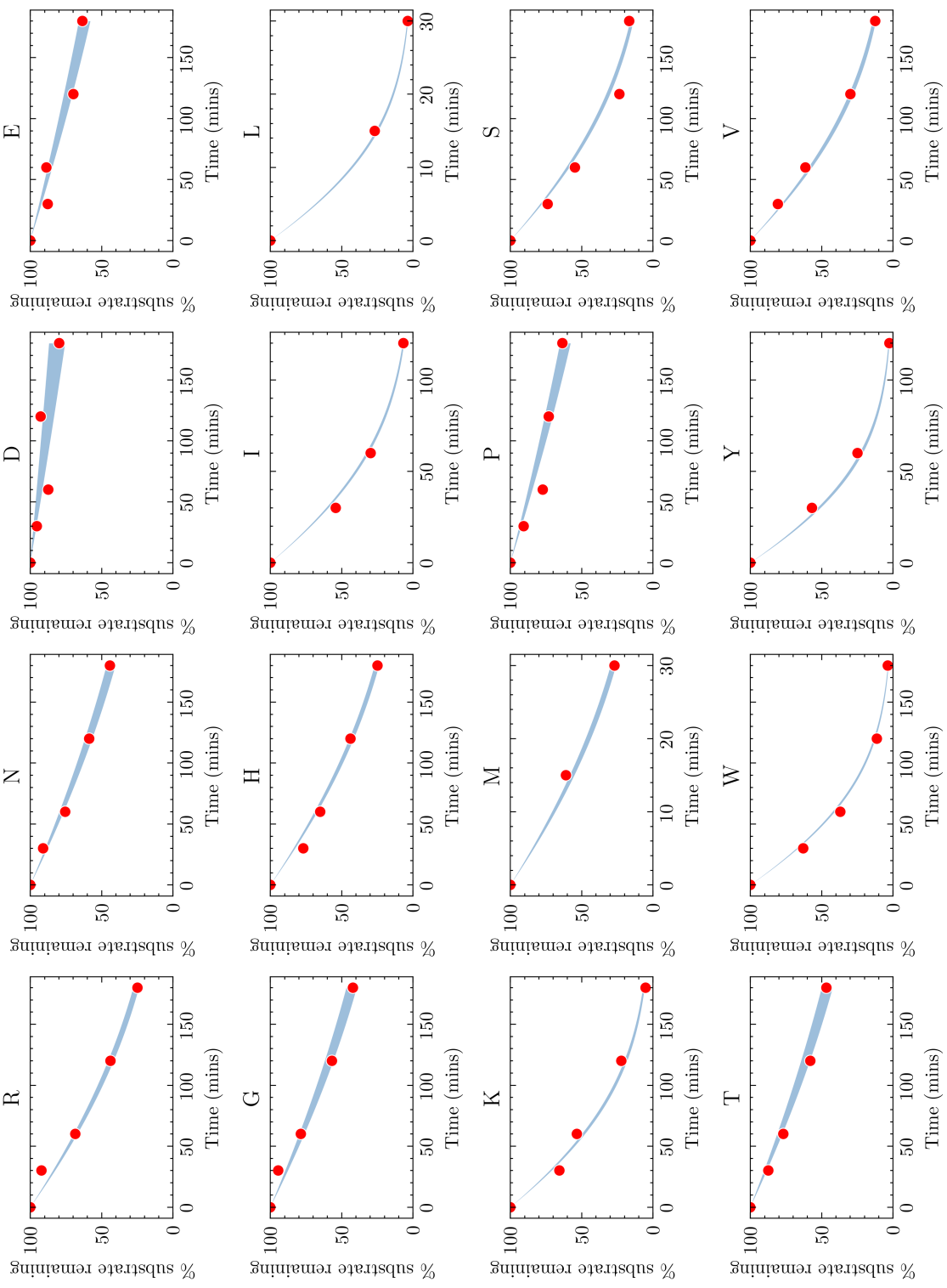


Figure 4.7. 90% credible interval shown from sampling 5,000 sets of parameters randomly from the posterior distributions for each of the 16 XS-L peptides.

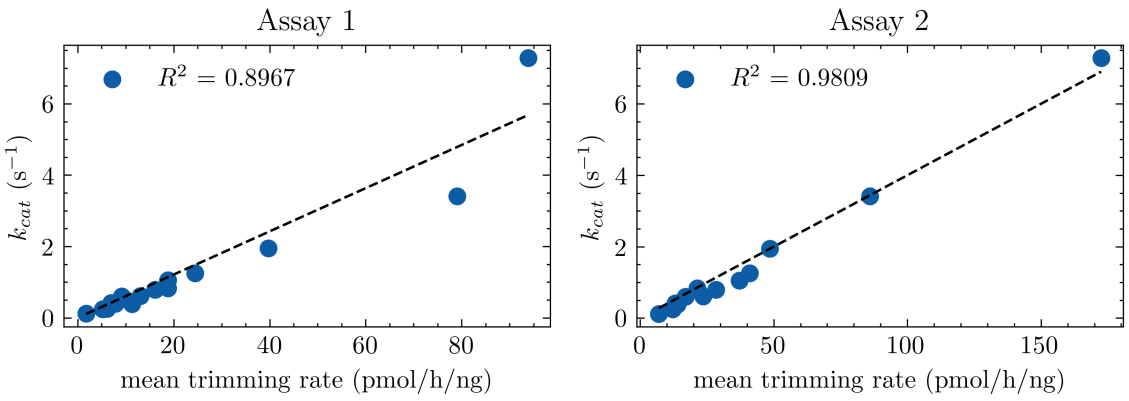


Figure 4.8. Catalytic rate plotted against mean trimming rate for the lower (Assay 1) and upper (Assay 2) limits of the error bars in the Hearn dataset [72]. Line of best fit from linear regression with 0 intercept is depicted for each assay, with goodness-of-fit shown by R^2.

a much stronger linear correlation between the inferred values of k_X and the data from the second repeat (corresponding to the higher mean trimming rates), implying that the time series data presented in Hearn et al.'s original paper corresponds to the data in Assay 2.

The line-of-best-fit between for Assay 2 the catalytic rate and the trimming rate is the line $y = 0.040x$. This gives a scaling factor of 0.040 for converting any predicted trimming rates to catalytic rates to be used with a Michaelis constant of 100μM.

4.4 Discussion

4.4.1 Amino acid encodings are problem-specific

In the previous chapter, when training a predictor of TAP binding affinity, models trained using the Feng or the Benner family of substitution matrices performed significantly worse than encodings such as sparse or BLOSUM matrices (Figure 3.5). When training our predictor of ERAP1 trimming, however, our optimal ensemble contained regressors trained using both the Feng and Benner74 matrices. Conversely, the sparse and physicochemical encodings were associated with low accuracy in the training of the ERAP1 model, having been towards the top performing encodings for the TAP model.

It could be argued that the sparse encoding performs worse on the ERAP1 problem because sparse encodings suffer from overfitting on smaller training sets. However, it is less intuitive why the Feng substitution matrix performs so poorly on the TAP problem relative to the ERAP1 problem. This highlights the difficulties associated

with selecting an encoding for the amino acids in a model a priori, and further justifies the systematic approach we took in our model creation.

4.4.2 Limitations

We found a weaker correlation between our ERAP1 model's predictions and the available data (Figure 4.5) than for *PanTAP* in the previous chapter (Figure 3.7). In this section we discuss possible factors underlying this performance, as well as other limitations with our model.

4.4.2.1 Small training data set

A clear limitation in this work is the low number and diversity of peptides in the training set. Of the 85 unique peptides in the training set, 45 contain the sequence SIINFEKL. With the large combinatorial space of possible substrates (20^n for an n-mer) and the importance of each residue, it would be highly surprising if such a small training set was sufficient to accurately learn the underlying rules of ERAP1's substrate specificity.

4.4.2.2 Possible length bias

Our final model appears to form less accurate predictions for 8mers than for longer peptides, although this difference was not statistically significant for the low number of examples in our training set. It is worth considering, however, that this may be caused by the underlying mechanism of peptide trimming by ERAP1.

Longer substrates are able to concurrently bind to the enzyme's regulatory site from their C-termini, causing an allosteric change and essentially self-activating ERAP1 for hydrolysis of the peptide's N-terminus. It has been reported (and can be established from crystal structures) that 8mers are below the minimum length required to stretch between the regulatory and active sites. Hence, the underlying mechanism of trimming of 8mers is likely to be different. In particular, one would expect the substrate's C-terminus amino acid to have a significantly reduced effect on activity than for longer peptides. Therefore, it may be more appropriate to train a separate predictive model for 8mer trimming prediction.

4.4.2.3 Application of in vitro to in vivo

We have trained a predictive model of ERAP1 activity by using only *in vitro* enzyme assays. In using this type of data to train our model, we are making the assumption that ERAP1's specificity *in vivo* can be predicted from *in vitro* behaviour.

In principle, this assumption appears justifiable: the binding of enzyme and substrate should follow the same physical laws in cells as it does in *in vitro*. However, the relative concentrations of ERAP1 and its various substrates will likely differ *in vivo* substantially from the *in vitro* conditions. This may result in a deviation from Michaelis-Menten kinetics (for instance, to substrate-inhibition kinetics, which have been reported anecdotally for certain ERAP1 allotypes [54]).

4.4.2.4 ERAP1 polymorphism

ERAP1 is a polymorphic gene, with 10 major allotypes covering over 94% of the global population (and >99.9% in the European population) [78]. A further 6 allotypes are particularly prevalent in the African population, covering over 20%. These single nucleotide polymorphisms (SNPs) can have significant effects on the enzyme's ability to trim certain substrates [78]. Our predictive model does not currently have the ability to predict these differences, primarily because it was not apparent which allotype had been used for each of the studies in our training set. To our knowledge, only one study of specific activity across different ERAP1 allotypes exists, examining the trimming of GLEQLESIINFEKL down to SIINFEKL through quantification of intermediate products [78]. In isolation, this study highlights the fact that differences between common allotypes exists but is insufficient to fully characterise this heterogeneity in specificity.

4.4.3 Future work

To address the issue of limited *in vitro* assay data, it may be possible to train a predictor of ERAP1 activity without carrying out time-consuming and expensive *in vitro* digestion assays. In the last few years, researchers have compared peptidomes from ERAP1 knockout (KO) cells to peptidomes of wild-type (WT) cells [164]. Comparing peptidomes from WT cells to ERAP1 KO cells could theoretically be used to infer the activity of ERAP1 *in vivo*.

There are some important caveats to this which make this task non-trivial. Firstly, peptidomic data is only semi-quantitative. The relative abundance of a peptide in one

peptidome versus another cannot easily be estimated. This could be very problematic for labelling negative cases (i.e. poor ERAP1 substrates).

To illustrate this, we can imagine an abundantly-presented peptide in the KO peptidome that is a good substrate for ERAP1. We would therefore expect a large drop in presentation in the WT peptidome, and this may well occur but would not necessarily be detectable through mass spectrometry. Labelling this peptide as a poor ERAP1 substrate would be a mistake, so we would need to exercise caution in how we construct our negative cases.

Looking at the problem from the opposite direction, peptides present in the WT peptidome but not the KO peptidome presumably had a precursor that could not be trimmed without ERAP1. However, it would be dangerous to assume which of the possible precursors of this peptide required trimming by ERAP1.

Hence, the only safe conclusion from comparing peptidomes would be that peptides on the cell surface on the KO cells but not the WT cells are likely to be good substrates for ERAP1.

4.4.4 Concluding remarks

In this chapter we have developed a regressive model to predict the rate at which ERAP1 trims peptides of varying length and composition. Although training data was extremely limited, our model's cross-validation performance gives us reason to be hopeful that it might generalise to a wider range of potential substrates.

We have demonstrated how the regressor's predictions can be used to derive Michaelis-Menten kinetic parameters. These parameters can then be used to extend the predictions of our model into more plausible contexts *in vivo*, where multiple substrates compete for binding to ERAP1 via adapted Michaelis-Menten kinetics for competitive inhibitors. This will then be integrated in to a mechanistic model in Chapter 5.

Chapter 5

Modelling of tapasin-assisted MHC-I loading

In this chapter, we adapt an existing systems biology model of peptide loading in the endoplasmic reticulum (ER). As well as re-fitting the model to align with justified assumptions of raw peptide-MHC (pMHC) numbers, we extend the model to incorporate the predictions of *PanTAP* (Chapter 3) and our ERAP1 prediction model (Chapter 4).

5.1 Introduction

Once an epitope or precursor has been translocated into the endoplasmic reticulum by TAP, it must be loaded onto MHC-I before it can be presented on the cell surface. For many alleles, this process would be prohibitively inefficient if it were not for the chaperone molecule tapasin [14, 76]. Tapasin stabilises the peptide loading complex (PLC) by bridging TAP and MHC-I [120, 133] and is known to influence the extent of peptide optimisation by steering the cell surface cargo towards peptides with low off-rates [76].

The extent of this enhancement is known to vary across different MHC class I alleles. Certain alleles (e.g. HLA-B*44:02) are highly dependent on tapasin for the efficient loading of peptides, whereas other alleles have been found to load peptide efficiently without the assistance of tapasin (e.g. HLA-B*44:05) [12, 14]. We describe this property as *tapasin dependence* and formally define it as the ratio of pMHC presented in the presence of tapasin to the level presented in absence of tapasin.

Bashirova et al. measured this quantity using a tapasin-deficient (.220) and reconstituted (.220tpn) human B-cell line for 97 HLA-A, -B and -C alleles. They found that tapasin dependence varied over two orders of magnitude and correlated negatively with the breadth of peptides presented [14]. This can have consequences for immunosurveillance, with tapasin independent alleles being associated with slower disease progression and lower viral load in HIV and Dengue virus infection. Tapasin is also often downregulated by tumours, presumably as a means of escaping immunosurveillance. Hence, it is essential to know or predict the tapasin dependence of any MHC-I allele if antigen processing and presentation is to be understood.

Although Bashirova et al.'s protocol can be easily applied to measure tapasin dependence of alternative MHC-I alleles, the highly polymorphic nature of MHC-I means that screening all possible allotypes will never be a viable strategy. Therefore, we are left with the task of predicting tapasin dependence for alleles outside of the 97 for which measurements are available.

In this chapter, we extend Dalchau et al.'s peptide loading model to include ERAP1 and TAP. We attempt to predict tapasin dependence for unknown alleles using the MHC-I sequence and predicted structure, and discuss why this is currently an elusive problem.

5.2 Methods

5.2.1 The Dalchau model

5.2.1.1 Model overview

The starting point for our model of peptide loading in the ER is a biological systems model of tapasin-assisted peptide loading presented by Dalchau et al. (henceforth referred to as the *Dalchau* model) [43]. Peptide is loaded onto MHC-I in the ER via either a tapasin-dependent or tapasin-independent pathway. The tapasin dependent pathway results in the intermediate formation of a complex consisting of tapasin, MHC-I and the peptide. Peptide dissociates from this complex at an enhanced rate, characterised by the scalar parameter q, relative to the tapasin-independent pathway.

This model has been validated on multiple levels using experimental data, including:

1. Quantitative presentation of SIINFEKL and 3 similar sequences [76].

2. Quantitative presentation of competing peptides, using intracellular measurements of peptide abundance [22].

3. Quantitative pulse-chase measurements of the biogenesis of tapasin dependent versus independent alleles [12].

For this final study, an extended version of the *Dalchau* model was used to investigate the molecular mechanism of MHC-I loading greater detail [12]. This helped to identify structural intermediates of peptide loading and assign their function in the peptide filtering process. For our purposes, the core model (shown in a schematic in Figure 5.1) is sufficient, but it should be noted that the peptide–MHC binding rates b and c are complex and are implicitly capturing a conformational change in MHC-I.

The system in the schematic consists of 10 reactions describing changes in the number of molecules of MHC class I (M), tapasin (T), peptide of species i (P_i), egressed MHC-I (Me), and complexes of these molecules:

$$\varnothing \underset{d_M}{\overset{g_M}{\rightleftharpoons}} M \qquad \varnothing \underset{d_T}{\overset{g_T}{\rightleftharpoons}} T \qquad \varnothing \underset{d_P}{\overset{g_i}{\rightleftharpoons}} P_i$$

$$M + T \underset{u_T}{\overset{b_T}{\rightleftharpoons}} TM \qquad TM + P_i \underset{q \cdot u_i}{\overset{c}{\rightleftharpoons}} TMP_i \qquad M + P_i \underset{u_i}{\overset{b}{\rightleftharpoons}} MP_i$$

$$TMP_i \xrightarrow{u_T \cdot v} T + MP_i \qquad MP_i \xrightarrow{e} MeP_i \qquad MeP_i \xrightarrow{u_i} Me$$

$$Me \xrightarrow{d_{Me}} \varnothing$$

Figure 5.2. Chemical reaction network describing the processes in the original *Dalchau* model.

5.2.1.2 Extension of Dalchau model

We adapted the *Dalchau* model to incorporate the predictions of TAP translocation and ERAP1 trimming arising from the regressive models in Chapters 3 and 4 respectively.

Firstly, TAP translocation was used to replace the supply rate of each peptide (g_i) to the ER. A new subcellular compartment (the cytosol) was added, in which peptides are produced and degraded, or translocated to the ER via the formation of an intermediate complex with TAP (S). The binding kinetics of this complex are determined

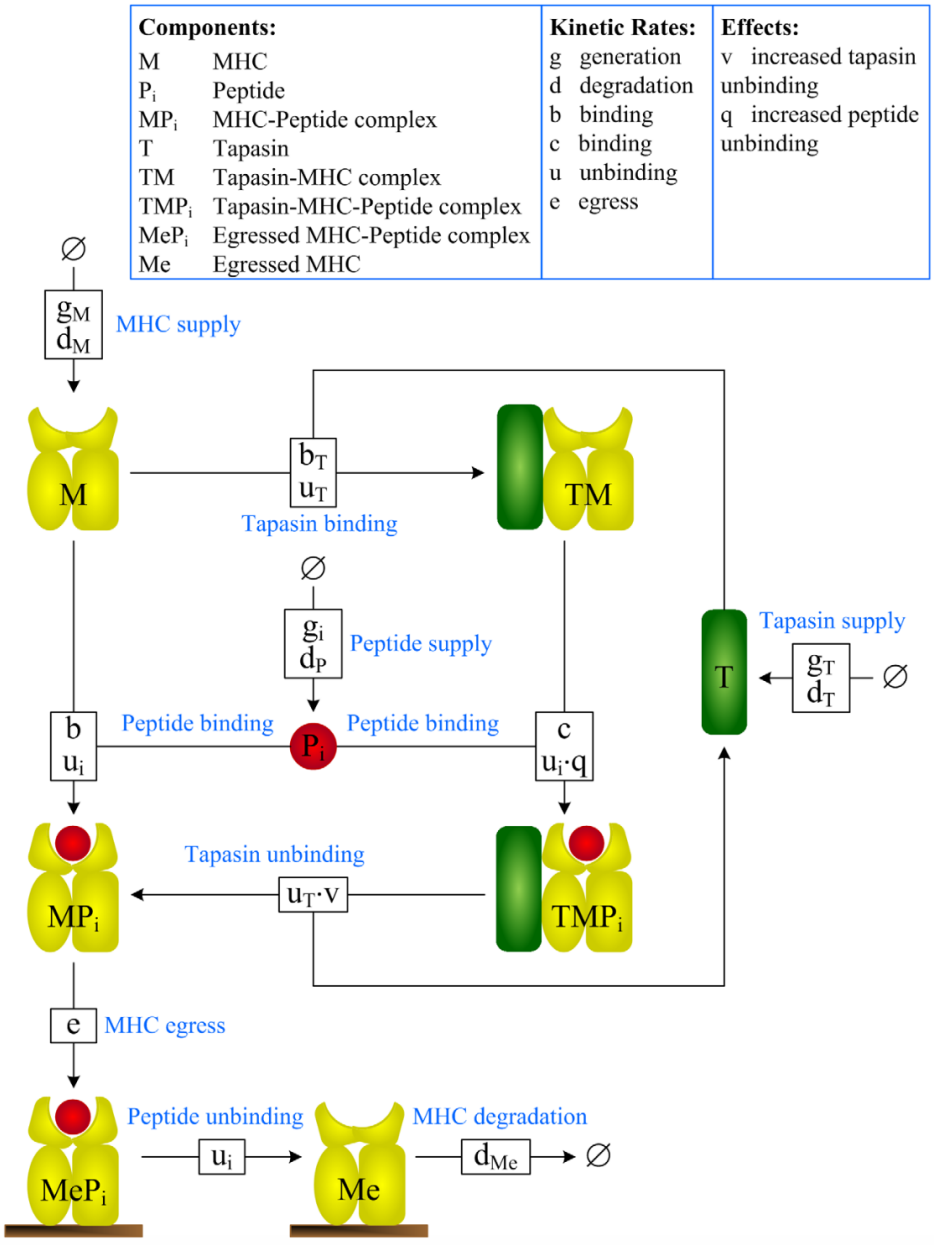


Figure 5.1. Schematic of original *Dalchau* model reproduced from [43]. Each box in the model represents a reaction, with inbound edges representing reactants and outbound edges representing products. Boxes are labeled with corresponding reaction rates, where a single rate denotes an irreversible reaction and two rates denote a reversible reaction. For reversible reactions, the rate of the forward reaction is indicated on top.

by the predicted binding affinity, K_D, returned by *PanTAP*. We assume that the forward rate, b_i, is substrate specific, whereas the peptide-TAP compelex dissociation rate, u_S, is assumed to be identical across all potential substrates. These are related to the predicted binding affinity through the equation:

$$K_D = \frac{u_S}{b_i}.$$

The reactions occurring in the new cytosolic compartment can therefore be written as:

$$\varnothing \underset{d_C}{\overset{g_i}{\rightleftharpoons}} P_i^C \qquad P_i^C + S \underset{u_S}{\overset{b_i}{\rightleftharpoons}} SP_i^C \qquad SP_i^C \xrightarrow{k_S} S + P_i^{ER}$$

where the superscripts, C and ER, are used to distinguish peptide in the cytosol from peptide in the endoplasmic reticulum respectively.

ERAP1 was permitted to trim only free (unbound) peptide in the ER. Although a mechanism of action consisting of MHC-I acting as a template for ERAP1 trimming has been proposed in the past [56], recent evidence suggests that the ERAP1 active site cannot access the N-terminus of MHC-I bound peptide [110, 111].

We incorporate ERAP1 trimming through the Michaelis-Menten equation with competitive inhibition. For a peptide, P_i, with predicted catalytic rate, k_i, the rate of trimming of the peptide's N-terminus amino acid by ERAP1, $r(\boldsymbol{P})$, is given by:

$$P_i \cdot r(\boldsymbol{P}) := \frac{\mathrm{d}P_i}{\mathrm{d}t} = -P_i \frac{E_0 k_i}{K_M + \sum_j P_j}, \tag{5.1}$$

where $\boldsymbol{P}$ is the state vector of the levels of all peptides resident in the endoplasmic reticulum and j is a dummy index used to sum over this state vector.

The revised chemical reaction network describing the newly extended *Dalchau* model is given in Figure 5.3 below and summarised by the schematic in Figure 5.4. By applying the Law of Mass Action, we can write this as a system of ODEs (given in Section A.1).

$$\varnothing \underset{d_C}{\overset{g_i}{\rightleftharpoons}} P_i^C \qquad P_i^C + S \underset{u_S}{\overset{b_i^S}{\rightleftharpoons}} SP_i^C \qquad SP_i^C \xrightarrow{k_S} S + P_i^{ER}$$

$$\varnothing \underset{d_M}{\overset{g_M}{\rightleftharpoons}} M \qquad \varnothing \underset{d_T}{\overset{g_T}{\rightleftharpoons}} T \qquad P_i^{ER} \xrightarrow{d_{ER}} \varnothing$$

$$M + T \underset{u_T}{\overset{b_T}{\rightleftharpoons}} TM \qquad P_i^{ER} \xrightarrow{r} P_{i-1}^{ER} \qquad TM + P_i^{ER} \underset{q \cdot u_i}{\overset{c}{\rightleftharpoons}} TMP_i$$

$$M + P_i^{ER} \underset{u_i}{\overset{b}{\rightleftharpoons}} MP_i \qquad TMP_i \xrightarrow{u_T \cdot v} T + MP_i \qquad MP_i \xrightarrow{e} MeP_i$$

$$MeP_i \xrightarrow{u_i} Me \qquad Me \xrightarrow{d_{Me}} \varnothing$$

Figure 5.3. Chemical reaction network for the *Dalchau* model with added TAP and ERAP1. P_{i-1} is used to denote the peptide P_i after the removal of a single amino acid from the N-terminus.

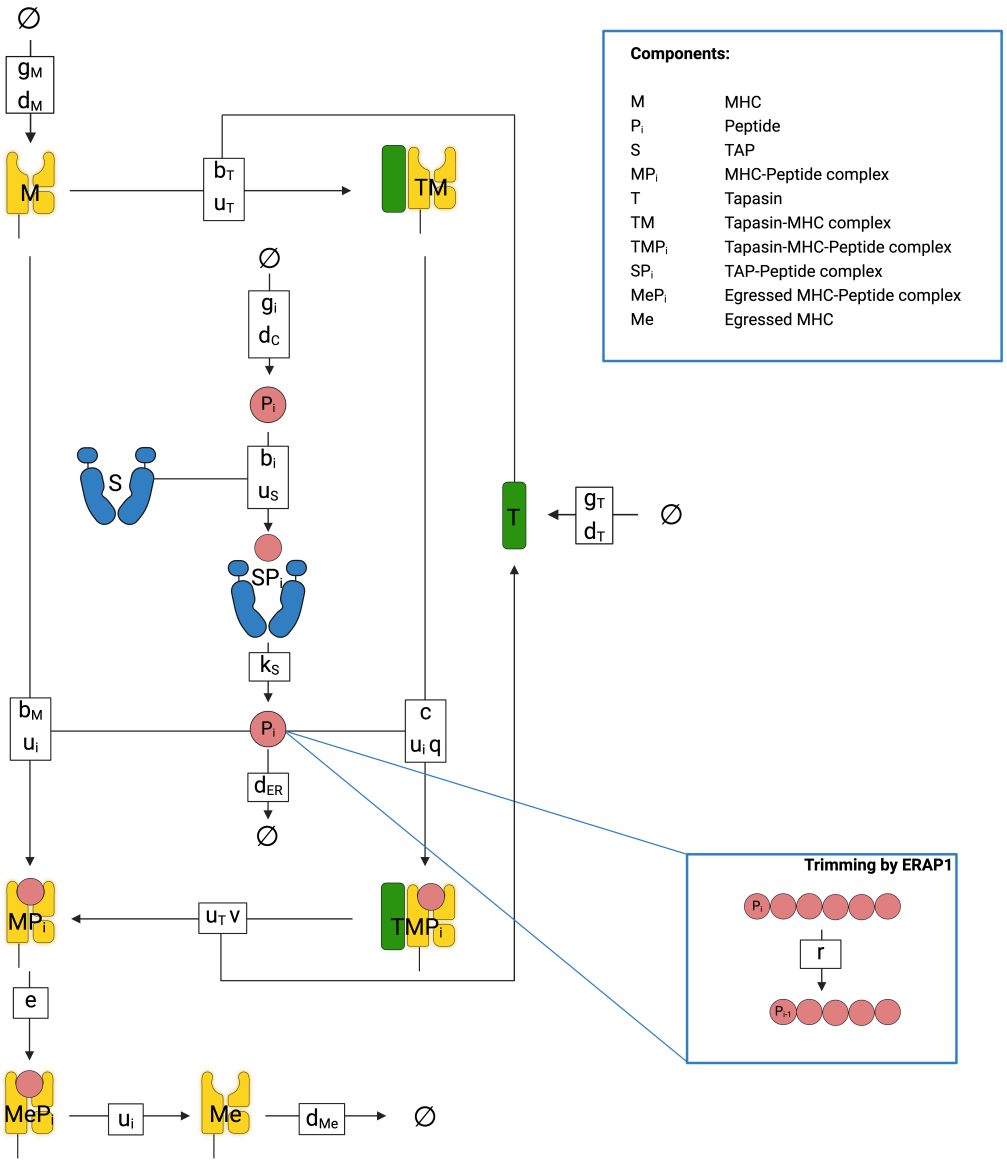


Figure 5.4. Extended *Dalchau* model with TAP and ERAP1. Each box in the model represents a reaction, with inbound edges representing reactants and outbound edges representing products. Boxes are labelled with corresponding reaction rates, where a single rate denotes an irreversible reaction and two rates denote a reversible reaction. For reversible reactions, the rate of the forward reaction is indicated on top. For simplicity, only the movement of a single species of peptide through the model is shown (denoted by P_i). The removal of a single amino acid from the N-terminus of this peptide by ERAP1 is shown in the callout box, yielding the species P_{i-1}.

5.2.2 Refitting the Dalchau model

5.2.2.1 Endogenous peptide

Endogenous supply is included in the model to compete with the target peptide for availability of the various molecules (e.g. TAP, ERAP1 and MHC-I). In the cytosol, we treat endogenous peptide as a single species with an associated binding rate, b_{self}, with TAP. This represents the weighted arithmetic mean of the individual binding rates of the many different peptides that make up the cytosolic endogenous peptide supply.

In absence of any target peptide, we can solve the system of equations in the cytosol given in Figure 5.3 to determine the rate of translocation of endogenous peptide to the ER at steady state. The rate of translocation is given by $k_S \cdot SP^C_{self}$, which is related to the rate through the equation

$$(u_S + k_S)\, SP^C_{self} = \frac{b_{self} \cdot \left(S_0 - SP^C_{self}\right)\left(g_{self} + u_S \cdot SP^C_{self}\right)}{d_C + b_{self} \cdot \left(S_0 - SP^C_{self}\right)},$$

where S_0 is the total number of TAP molecules in the cell. We use John Yewdell's estimates to assume that S_0 is 10,000 molecules, g_{self} is 2×10^6 peptides per second, and the total rate of translocation of endogenous peptide to the ER is 30,000 peptides per second [169]. We also assume that the turnover rate of TAP, k_S, is 5 s^{-1} [2]. Denoting the rate of translocation of endogenous peptide at equilibrium as $\gamma := k_S \cdot SP^C_{self}$, we can solve this equation for b_{self} to give:

$$b_{self} = \frac{d_C \cdot (u_S + k_S)}{(g_{self} - \gamma) \cdot (k_S \cdot S_0/\gamma - 1)}. \tag{5.2}$$

Hence, b_{self} can be set for any d_C and u_S to give an endogenous peptide supply to the ER that is consistent with Yewdell's estimate.

Once translocated to the ER, we separate the endogenous peptide into two species: strong and weak/non-binders to MHC-I. The proportion of strong binders in the total supply of peptides translocated by TAP is denoted by $\rho_b \in (0, 1)$, with the remaining peptides assumed to be weak/non-binders.

Although many endogenous peptides will be trimmed by ERAP1 in the ER, we assume that the majority of this trimming results in the production of a new endogenous peptide, hence having no effect on the size of the endogenous peptide population.

This assumption is justified by the length preference of ERAP1 for peptides longer than 9 amino acids [35]. It is highly likely that weak/non-binders will be trimmed to form strong binders through the creation of peptides of a more appropriate length for fitting into the MHC-I binding groove. However, in the interest of developing a parsimonious model, we assume that this behaviour will be captured in the parametrisation of the strong binder proportion, ρ_b. In summary, we only consider endogenous peptide as a source of competition for binding to ERAP1 through the term in the denominator of Equation 5.1.

5.2.2.2 Conversion of concentrations

PanTAP is trained to return a predicted binding affinity in units of nanomolarity, and the Michaelis constant for ERAP1 was previously set at 100 micromolar. As we wish to use the extended *DaIchau* model to predict raw numbers of peptide-MHC complexes on the cell surface, we must convert between units of molarity and number of molecules per compartment.

To convert the Michaelis constant, we assume that the volume of the ER is 10% of the total cell volume [42], and that the typical cell volume is approximately 4,500 μm^3 [41]. This results in the following calculation of K_M in terms of molecules per compartment:

$$K_M = \underbrace{100 \times 10^{-6}}_{\text{conversion } \mu\text{M to M}} \times \underbrace{0.1}_{\text{10\% total volume}} \times \underbrace{4.5 \times 10^{-12}}_{\text{4,500 } \mu\text{m}^3 \text{ in litres}} \times \underbrace{6.022 \times 10^{23}}_{\text{Avogadro's constant}} .$$

We also assume that the volume of the cytosol is similar to the measured volume of the cytosol in a HeLa cell, reported as 940 μm^3 [61]. Hence, to convert the predicted peptide-TAP binding affinity from nM to molecules per compartment, we multiply the affinities by the scaling factor, β, defined by

$$\beta = \underbrace{9.4 \times 10^{-13}}_{940\mu\text{m}^3 \text{ in litres}} \times \underbrace{10^{-9}}_{\text{conversion nM to M}} \times \underbrace{6.022 \times 10^{23}}_{\text{Avogadro's constant}} .$$

5.2.2.3 Assumptions of peptide-MHC abundance

The *DaIchau* model was originally parametrised using observations from various *in vitro* assays using H2-Kb transfected .220 cells. Throughout these experiments,

surface abundance of MHC-I or specific peptides was recorded through the mean fluorescence intensity (MFI) observed by the binding of specific antibodies to pMHC complexes.

Fluorescence data only gives us a semi-quantitative measurement of the surface peptide-MHC abundance. Although an assumption of linearity is made regarding the fluorescence (i.e. an MFI of 2,000 corresponds to twice as many pMHC complexes as an MFI of 1,000), the raw number of complexes can not be determined without proper calibration.

A consequence of using semi-quantitative data for parameter estimation is that certain parameters may not be uniquely identifiable. For example, the supply rate of peptide to the endoplasmic reticulum can be shown analytically to scale linearly with the level of surface pMHC, so cannot be accurately determined unless raw numbers are estimated.

To resolve this and ensure consistency of predictions with immunological understanding, we made three assumptions about the raw number of molecules at different stages of the model:

1. The cell presents between 100,000 and 200,000 total molecules of each MHC-I allotype on the surface at steady-state, based on common estimates in the literature and specific mAb measurements for H2-Kb [5, 145].

2. The total supply rate of endogenous peptide to the endoplasmic reticulum is 30,000 peptides per second (assuming 3 peptides translocated per TAP molecule per second and 10,000 TAP molecules per cell, as estimated by John Yewdell) [169].

3. At steady-state, the cells present approximately 65,000 to 85,000 copies of SIINFEKL. This estimate is based on the 67,072 and 84,664 SIINFEKL-Kb complexes quantified by Porgador et al. on the surface of a H-2Kb transfected L929 cells expressing a minigene containing SIINFEKL with a signal sequence and methionine leader sequence respectively[128].

Although these are rough estimates, based on old experimental data, often from different cell lines and species (e.g. L929, which is a mouse fibroblast cell line), we theorised that significant error in these estimates would result in us being unable to fit our mechanistic model to the experimental data.

5.2.2.4 Parametrisation dataset

To re-parametrise the *Dalchau* model, we used the same measurements of the processing and presentation of 4 variants of SIINFEKL used to parametrise the original version of the model [43]. The authors studied the H2-Kb transfected human B cell line .220 (which does not naturally express tapasin) and a tapasin-reconstituted .220 cell line (.220tpn). Peptides were expressed in the cells as minigenes encoding the octameric peptide preceded by a methionine, which is expected to be cleaved by cytosolic aminopeptidase activity [76]. Surface peptide abundance was measured by flow cytometry using the 25.D1 antibody (which binds to SIINFEKL and SIINFEKL variants) and the Y3 antibody (which binds to empty and occupied H2-Kb complexes).

The data can be broken down into 3 sections, all measured in the tapasin-positive and tapasin-deficient cell lines:

1. Cell peptidome stability measured following cell treatment by brefeldin A (BFA).
2. SIINFEKL and SIINFEKL variant (collectively denoted SIINXEKX) presentation in untreated cells.
3. MHC-I trafficking efficiency measured by endoglycosidase-H (EndoH) resistance in a pulse-chase assay.

BFA treatment prevents the fusing of the Golgi-apparatus to the endoplasmic reticulum, cutting off any additional MHC-I from reaching the cell-surface. This was used to measure the stability of the repertoire of pMHC complexes on the cell surface by quantifying with the Y3 antibody for 21 hours post BFA treatment.

SIINXEKX peptides were introduced separately to the two cell lines using minigenes encoding the sequence MSIINXEKX. After a long period of time (unspecified by *Dalchau* et al but assumed to be 48 hours), the level of each peptide on the cell surface was evaluated using flow cytometry with 25.D1 antibodies.

Endoglycosidase-H is an enzyme that cleaves glycosylated proteins. In the ER, MHC-I contains an N-linked glycan, so is sensitive to EndoH. This glycan is removed when the MHC-I is trafficked to the cell surface by the Golgi apparatus. In the pulse-chase assay, an initial population of MHC-I in the ER was radiolabelled and the rate at which this population became EndoH resistant (i.e. moved to the cell surface) was tracked over the course of 7 hours.

5.2.2.5 Simulation of experimental data

We computationally simulated three different experiments in order to reproduce the three studies described in the previous section. We used the Law of Mass action to write the extended *Dalchau* model as a system of ordinary differential equations (ODEs).

To simulate the BFA assay, we first set all initial states to 0 (except for TAP, which was initialised at $S(0) = 1 \times 10^4$ molecules) and solved the system of ODEs in A.1 up to time $t = 1 \times 10^8$ seconds (chosen to try to get close to the system's equilibrium point) using the LSODA solver in SciPy's suite of numerical differential equation solvers [160]. We set the supply rate of the target peptide, P_i, to the cytosol to 0 since no target peptide was supplied for the BFA decay assay. The system state at the final time point was then used as a start point to find the exact equilibrium points of the system by setting all derivatives to 0 and using SciPy's *fsolve* function [160]. The equilibrium points of the tapasin deficient system (.220 cells) were found in exactly the same way by setting the tapasin supply rate, g_T, to 0. In this way, we obtained the equilibrium solutions, $\mathbf{X}^{*TPN}$ and $\mathbf{X}^{*KO}$, where $\mathbf{X} = \left[P^C_{self},\ S,\ SP^C_{self},\ ...,\ Me\right]$ is the system state vector and the superscripts TPN and KO and denote the .220tpn and .220 cell lines respectively.

We used these equilibrium solutions as the startpoint to simulate a system of ODEs (Equations 5.3-5.5) describing the dissociation of endogenous peptide-MHC complexes and empty MHC from the surface following the inhibition of further egress from the ER:

$$\frac{\mathrm{d}MeP_b}{\mathrm{d}t} = -u_b \cdot MeP_b, \tag{5.3}$$

$$\frac{\mathrm{d}MeP_n}{\mathrm{d}t} = -u_n \cdot MeP_n, \tag{5.4}$$

$$\frac{\mathrm{d}Me}{\mathrm{d}t} = u_b \cdot MeP_b + u_n \cdot MeP_n - d_{Me} \cdot Me. \tag{5.5}$$

This system of ODEs was solved numerically for times between 0 and 21 hours. The proportion of total MHC-I remaining was then calculated by:

$$f_{BFA}(t) = \frac{MeP_b(t) + MeP_n(t) + Me(t)}{MeP_b(0) + MeP_n(0) + Me(0)}, \tag{5.6}$$

for $t \in \{1\text{h}, 3\text{h}, 6\text{h}, 21\text{h}\} := t_{BFA}$.

To simulate the presentation of the SIINXEKX peptides, the system of ODEs in A.1 was initialised at the equilibrium solutions $\mathbf{X}^{*TPN}$ or $\mathbf{X}^{*KO}$, depending on whether the .220tpn or .220 cell line was being simulated. The target peptide supply rate, g_i, was then set to non-zero and the system simulated for 48 hours.

The number of target peptide-MHC complexes at 48 hours, MeP_i, was taken and re-scaled to convert into units of fluorescence intensity. As a scaling factor, we used the proportionality constant, α^*, derived by *DaIchau* et al to minimise the normalised sum of squares error between the model predictions, MeP_i, and the observed data, y_i:

$$\alpha^* = \frac{\sum_{S\in\{TPN,KO\}} \sum_{i\in\text{peps}} MeP_i^S}{\sum_{S\in\{TPN,KO\}} \sum_{i\in\text{peps}} \left(MeP_i^S\right)^2 / y_i}, \tag{5.7}$$

where i is a dummy index looping over the 4 different experimental measurements of SIINXEKX abundance and the superscript S differentiates the .220tpn cells from the .220 cells. Hence, all model predictions are converted to predicted MFIs by the formula $f_{MFI} := \alpha^* MeP_i$.

Finally, we simulated the EndoH resistance assay. To do so, we adapted the extended *DaIchau* model to include two species of MHC-I — one radiolabelled and one unlabelled. The system was simulated using the equilibrium solutions as initial conditions. During the first 30 minutes of simulation, all MHC-I was added to the radio-labelled population. After this time, all newly added MHC-I to the system was added to the unlabelled MHC-I population. The system was simulated up to 7 hours following the completion of radio-labelling.

The proportion of EndoH resistant radio-labelled MHC-I in the model was then calculated at various times according to the formula:

$$f_{EndoH}(t) := \frac{Me_{all}(t)}{M(t) + TM(t) + MP_b(t) + MP_n(t) + TMP_b(t) + TMP_n(t) + Me_{all}(t)}, \tag{5.8}$$

where

$$Me_{all}(t) := MeP_n(t) + MeP_b(t) + Me(t),$$

for $t \in \{0, 30, 60, 120, 210, 420 \text{ mins}\} := t_{EndoH}$.

5.2.2.6 Bayesian inference via MCMC

During the process of parameter determination, we aimed to identify parameters that could not be uniquely determined using the available data. In order to detect such non-identifiable parameters, we used a Bayesian inference approach to sample the inferred probability distributions of the model parameters using a Markov Chain Monte Carlo (MCMC) method.

As Bayesian inference is a computationally expensive method, we aimed to keep the number of parameters requiring fitting to a minimum. In order to achieve this, we fixed 8 parameters in the extended *Dalchau* model at the values estimated by Dalchau et al. or derived through their original fitting of the model. These parameters are listed in Table 5.1 with their associated values. The remaining model parameters were either not used in the original *Dalchau* model (e.g. total ERAP1 molecules in the ER, E_0) or were found to be incompatible with the extended model by resulting in poor fits to the experimental data.

For each candidate set of parameters, θ, the BFA decay, single peptide and EndoH resistance assays were simulated, yielding predictions $f_{BFA}(t, \theta)$, $f_{MFI}(\theta)$ and $f_{EH}(t, \theta)$ respectively. In addition to this, we also determined the predicted total H2-Kb on the cell surface at equilibrium in the .220tpn cell line, $f_{surf}(\theta)$, and the number of molecules of SIINFEKL presented in the .220tpn cell line, $f_{SL}(\theta)$.

For data corresponding to a measurement of fluorescence (e.g. the level of SIINFEKX measured using 25.D1), we assumed that the observable, y, was Gaussian

Symbol	Value	Units
g_T	1505	molecules s^{-1}
b_T	1.663×10^{-9}	molecules^{-1} s^{-1}
u_T	1.185×10^{-9}	molecules^{-1} s^{-1}
d_P	0.13	s^{-1}
d_T	1.726×10^{-3}	s^{-1}
d_M	7.989×10^{-5}	s^{-1}
v	936.3	
q	2.104×10^{4}	

Table 5.1. Summary of parameters fixed throughout model inference to original *Dalchau* values.

distributed around the model predictions, f, with a standard deviation linearly proportional to the model prediction. We assumed experimental noise of 10% for the f_{MFI} data and 25% for the f_{SL} and f_{surf} data — the latter aiming to capture our uncertainty in the estimation of the raw numbers of pMHC complexes.

For the BFA assay, each data point is calculated by dividing surface pMHC abundance by initial pMHC abundance. We assumed that the measurement error was Gaussian distributed, so the error in the percentage of initial pMHC remaining will be approximately Cauchy distributed (since the ratio of two Gaussian distributions is Cauchy distributed). Similarly for the EndoH assay, each point corresponds to the proportion of resistant MHC-I, so is again the ratio of two Gaussian distributions. We therefore assumed that the observable was Cauchy distributed around the model predictions. For both assays, we assumed that the Cauchy distribution had an associated scale parameter of 0.1.

Our log-likelihood function can therefore be written:

$$\begin{aligned}\ell(\theta) = &-\log 0.1 - \log \pi - \sum_S \sum_{t \in t^S_{BFA}} \log(1 + \left(f^S_{BFA} - y^S_{BFA}\right)^2 / 0.1^2) \\ &- \sum_S \sum_{t \in t_{EndoH}} \log(1 + \left(f^S_{EndoH} - y^S_{EndoH}\right)^2 / 0.1^2) - 5 \log 2\pi \\ &- \sum_S \sum_{i \in \text{peps}} 0.1 f^S_{MFI} - 50 \sum_S \sum_{i \in \text{peps}} \frac{(y^S_{MFI} - f^S_{MFI})^2}{\left(f^S_{MFI}\right)^2},\end{aligned} \tag{5.9}$$

where the sum over S sums the data corresponding to the .220tpn (TPN) cells and the .220 cell line (KO).

We used a uniform log prior to initially assign an equal probability of sampling any value between the established plausible parameter ranges. We initialised 3 Markov chains at random states sampled from the prior and ran them using the Haario Bardenet ACMC algorithm. After 350,000 iterations, we confirmed that the 3 chains had converged to their posterior distributions by ensuring that the convergence criterion, $\hat{r}$, was less than 1.05 for all parameters, and by inspection of the trace plots.

5.2.3 Tapasin dependence in the Dalchau model

We define the tapasin dependence of an MHC-I allele as the ratio of total MHC-I on the cell surface at equilibrium in the presence of tapasin to the total MHC-I on the cell surface at equilibrium in the absence of tapasin. As tapasin dependence is a property of individual MHC-I alleles and the *Dalchau* model was re-parametrised only for H2-Kb, we wanted to investigate which parameters should be altered when considering different alleles in order to change the tapasin dependence of the model.

Logically, one could change the tapasin dependence predicted by the *Dalchau* model by adjusting one of:

1. The availability of tapasin
2. The enhancement to binding rates associated with tapasin
3. The peptide-MHC binding rate in absence of tapasin

However, highly tapasin dependent alleles are characterised by lower pMHC numbers in absence of tapasin compared to tapasin independent alleles (rather than higher pMHC numbers in the presence of tapasin) [14, 167]. Whilst the first two parameters can increase the level of improvement in peptide loading efficiency created by tapasin, only changes in the third parameter value can affect the MHC-I presentation in the absence of tapasin.

Additionally, we found that by changing the other parameters the performance benefit of tapasin eventually saturates (as surface MHC-I eventually becomes limited by MHC-I supply, restricting the increase in presentation caused by tapasin). This saturation point occurs at an insufficient level to explain the upward of 200-fold increase in presentation observed in highly tapasin dependent cells [14].

Hence, we concluded that the level of tapasin dependence in the *Dalchau* model should be tuned through the peptide-MHC binding rate. In other words, this parameter should be MHC-I allele-specific.

We varied the peptide-MHC binding rate between 10^{-12} and 10^{-6} s^{-1} $molecules^{-1}$ and simulated the *Dalchau* model to equilibrium. The equilibrium states were then used to calculate the tapasin dependence produced by that binding rate, thus empirically deriving the relationship between binding rate and tapasin dependence. This permits the fixing of the binding rate for any MHC-I allele for which the tapasin dependence is known.

5.2.4 Prediction of tapasin dependence

Tapasin dependence has been measured and reported for 97 different HLA alleles. So that the *Dalchau* model might be applied more generally, we investigated whether this data set could be used to train a predictive model capable of accurately predicting tapasin dependence for other HLA and non-human MHC-I alleles.

We postulated that this quantity could be predicted from the sequence of the MHC-I allele. This assumption was supported by the fact that the MHC-I sequence has been used to predict pan-allelic differences in binding affinity through the pseudosequence used by *NetMHCpan* [114]. We therefore investigated whether we could train a regressive model using the Bashirova dataset with predictive capabilities.

5.2.4.1 Measurements of tapasin dependence

Bashirova et al. reported measurements of the tapasin dependence of 97 different HLA alleles [14]. To estimate tapasin dependence, the authors measured HLA class I surface expression levels in the human B cell line .220 (which does not naturally express tapasin) and a tapasin-reconstituted .220 cell line (.220tpn) using an anti-FLAG mAb. The tapasin dependence was reported as the ratio of mean fluorescence intensity (MFI) in .220tpn versus .220. Bashirova et al. discovered that tapasin dependence varied over more than 2 orders of magnitude across common HLA alleles. High tapasin dependence was associated with lower levels of MHC-I in the tapasin-deficient .220 cell line relative to less tapasin deficient alleles.

5.2.4.2 MHC-I sequence processing

For each of the 97 alleles, we exported the reference sequence from the IMGT database. This produced a length 182 sequence of amino acids. Typical encodings convert each amino acid to a vector of length 20 or more, so using the entirety of this sequence would make training a downstream model inefficient. It is also likely that the majority of the sequence does not contain information that is relevant to the prediction of tapasin dependence.

We therefore used three different representions of MHC-I sequence to train our regressive models:

1. Using the whole sequence of 182 amino acids.
2. Using the length 34 pseudosequence proposed by Nielsen et al. [114]
3. Using the 10 residues from the F pocket of MHC-I, since this region has been observed to be significant in tapasin binding [4, 64, 113].

5.2.4.3 Amino acid encoding

In addition to the substitution matrices, physicochemical features and sparse encoding described in previous chapters, we included a latent representation of the input sequence using *ESM-2* [101]. *ESM-2* is a transformer protein language model trained on protein sequences from the UniRef database and has been demonstrated to capture sufficient information to enable accurate atomic-level predictions of protein structure.

As tapasin dependence has been noted to correlate with structural properties of MHC-I such as conformation flexibility [64], we predicted that *ESM-2*'s learnt protein structure information might be beneficial to the downstream task of tapasin dependence prediction.

To use *ESM-2*, we passed each of the 97 allele sequences through the pre-trained 650 million parameter version of the model and concatenated the latent vectors from the final transformer layer to produce an encoding of each allele.

5.2.4.4 MHC-I structure prediction and encoding

We also investigated whether bringing information from the predicted structures of MHC-I alleles could enhance performance. To do so, we folded each allele using DeepMind's *AlphaFold2* model with the default hyperparameters [28].

AlphaFold returns predicted protein structures as sets of 3D coordinates of the protein backbone. These sets of coordinates cannot be passed to a regressor without further pre-processing, since our regressors expect a 1 dimensional vector of input features. We therefore pass each structure through *Foldseek* — a vector quantised variational autoencoder that uses protein bond and dihedral angles to reduce a 3D structure into a sequence of tokens corresponding to a 'structural vocabulary' of dimension 20 [158].

The overall effect is that the length 182 sequence of amino acids for each allele is converted to a length 182 sequence of structural tokens, capturing structural information without increasing the dimensionality of the input features. We then encoded this sequence by using the substitution matrix provided by *Foldseek* to convert the structural tokens to length 20 vectors of integers.

5.2.4.5 Regression training

As our prediction target, we log-scaled the mean tapasin dependence of the 97 alleles in the Bashirova dataset. For each encoding, we trained a support vector regression model (SVR) using 10-fold cross-validation to estimate performance on unseen data. Regressor hyperparameters were tuned using scikit-learn's *RandomizedSearchCV* with the same hyperparameter search space used in the training of *PanTAP* (Table 3.2).

5.3 Results

5.3.1 Extended Dalchau model inference

MCMC was used to infer the posterior distributions of the 10 fitted parameters from the *Dalchau* experimental data. The convergence of the 3 Markov chains and the resulting posterior distributions are shown in Figure 5.5.

The trace plots (shown on the right-hand side) show efficient convergence of the 3 chains from different initial states after approximately 50,000 iterations. Certain parameters converged much faster than this (e.g. the MHC-I supply rate to the ER, g_M), suggesting that the experimental data placed tighter constraints on these parameters than others.

The range of the x-axis for the posterior distributions corresponds to the range of the uniform prior used in the sampling, highlighting the difference between the prior and posterior distributions. For the majority of parameters, the chains converged to a tightly distributed posterior, indicating identifiability of the corresponding parameter from the data. However, three parameters stand out for having broad posterior distributions. The peptide-MHC egress rate, peptide-TAP dissociation rate and the total number of ERAP1 molecules in the ER all have associated posteriors spanning many orders of magnitude, showing that these parameters were not identifiable from the *Dalchau* data.

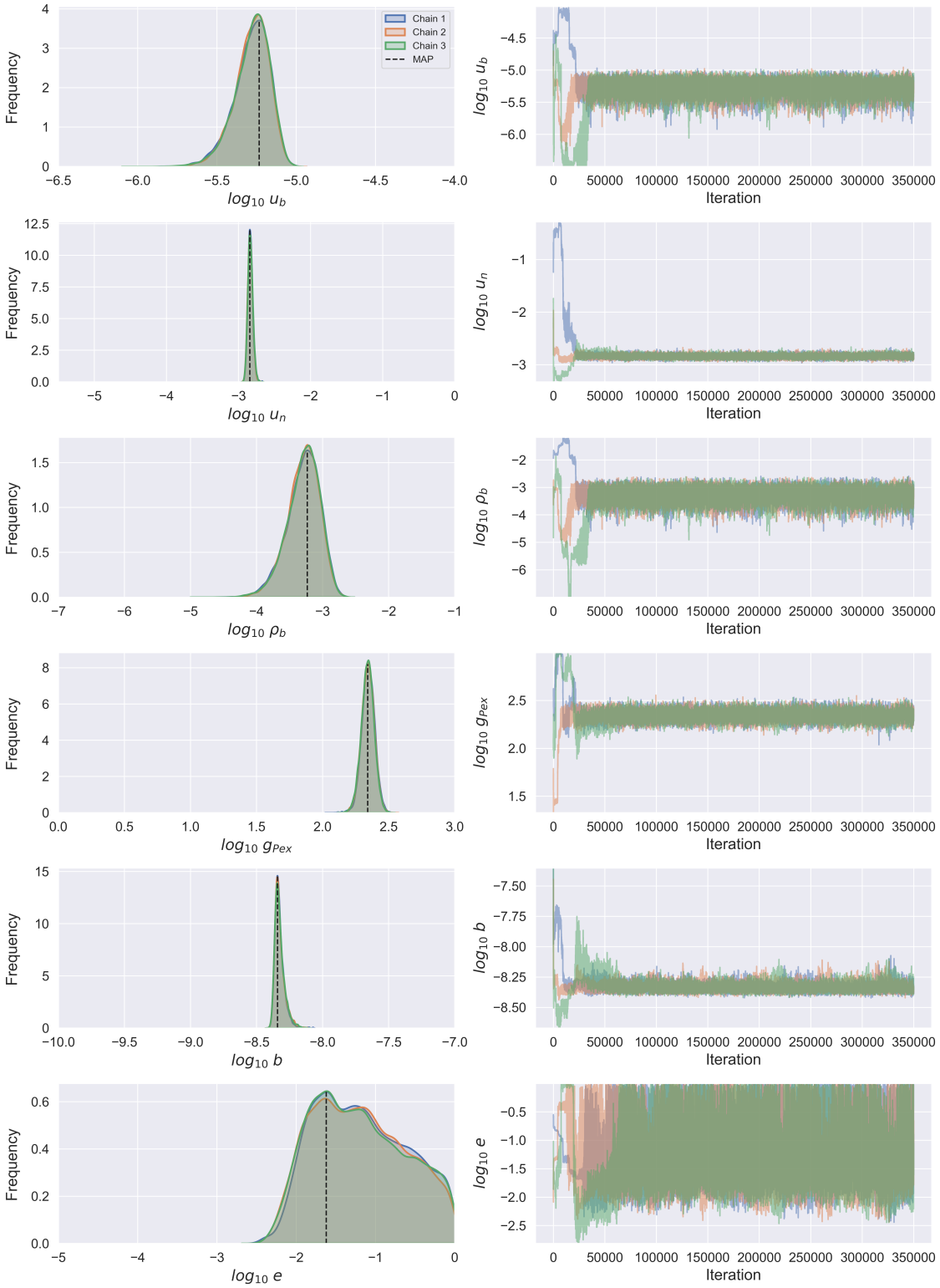

Chain 1
Chain 2
Chain 3
MAP
Frequency
$log_{10}\ u_b$
$log_{10}\ u_n$
$log_{10}\ \rho_b$
$log_{10}\ g_{Pex}$
$log_{10}\ b$
$log_{10}\ e$
Iteration

Figure 5.5

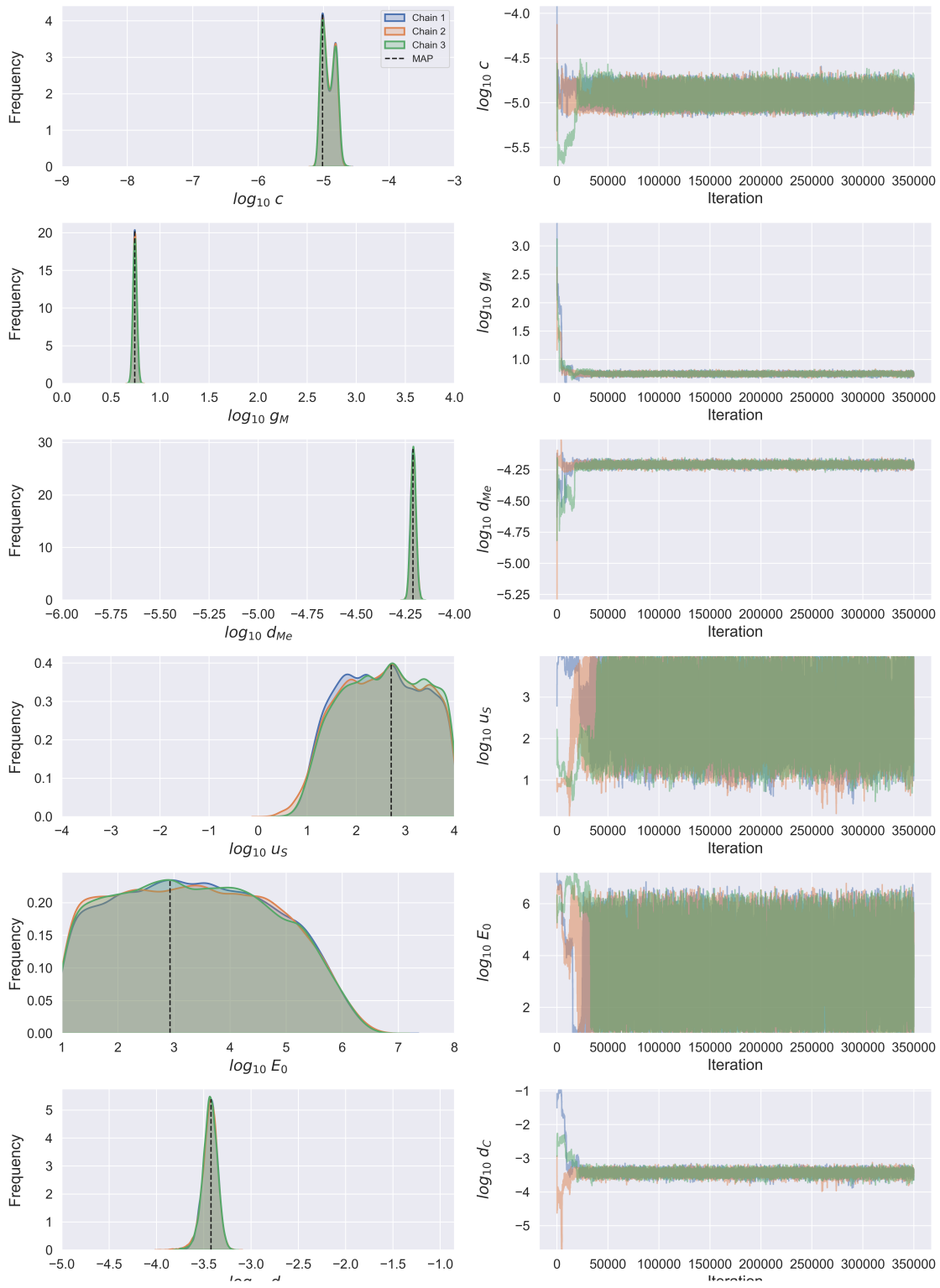


Figure 5.5. Posterior distributions (left) and trace plots (right) showing result of MCMC for the 10 inferred parameters in the extended *Dalchau* model. Posterior distributions were calculated from the 3 Markov chains after discarding the first 75,000 samples (the burn-in period, deduced from the trace plots) using the Seaborn library kernel density estimate (*kdeplot*) with default parameters. The maximum a posteriori (MAP) parameters are indicated by a dashed line for each posterior parameter distribution.

Parameter	Lower	Upper	MAP	Std.	ESS	$\hat{r}$
$\log u_b$	−6.5	−4.0	−5.23	0.11	6433.32	1.00
$\log u_n$	−5.5	0.0	−2.84	0.03	3072.39	1.00
$\log \rho_b$	−7.0	−1.0	−3.23	0.25	6304.38	1.00
$\log g_{Pex}$	0.0	3.0	2.34	0.05	3163.57	1.00
$\log b$	−10.0	−7.0	−8.34	0.04	1148.60	1.00
$\log e$	−5.0	0.0	−1.62	0.58	2036.37	1.00
$\log c$	−9.0	−3.0	−5.02	0.10	6320.75	1.00
$\log g_M$	0.0	4.0	0.74	0.02	7923.86	1.00
$\log d_{Me}$	−6.0	−4.0	−4.21	0.01	8096.70	1.00
$\log u_S$	−4.0	4.0	2.71	0.83	3585.62	1.00
$\log E_0$	1.0	8.0	2.93	1.37	7264.77	1.00
$\log E_0$	−6.5	−4.0	−3.42	0.08	4450.89	1.00

Table 5.2. Summary statistics from Bayesian inference. Lower and upper limits of log-scaled parameters were used to shape uniform log priors. Maximum a posteriori (MAP) parameter values are shown following 350,000 iterations. Standard deviation (std.) of posterior provided for each parameter. Effective sample size (ESS) and convergence criterion, $\hat{r}$, also provided to show convergence of chains.

The summary statistics associated with the MCMC experiment and Figure 5.5 are shown in Table 5.2, calculated assuming a burn-in period of 75,000 iterations.

To illustrate the consistency of the sampled parameters and extended *Dalchau* model with the experimental data, we sampled 1,000 sets of parameters from the posterior distributions and used these to simulate the experimental data in Figure 5.6. The model predictions show high consistency with the results of the BFA decay assay, both with and without tapasin, indicating that the stability differences in the peptidomes of both cell lines could be accurately represented by just two different species of endogenous peptide. The model is also highly consistent with the EndoH assay, although the predictions for the .220tpn cell line are more varied across the sampled parameters. We believe that this is due to the anomalously low data point at 210 minutes.

The mean predicted total surface MHC-I was between 160,000 and 170,000 complexes, whilst the predicted number of presented SIINFEKL peptides was just below 65,000. Both figures and distributions are consistent with the assumptions made in 5.2.2.3.

The predicted presentation of the SIINXEKX peptides is shown in Figure 5.6b. The

extended *Dalchau* model predictions agree closely with the experimental data in the .220tpn cell line, with the only small discrepancy occurring for the SIINYEKL peptide (which has the highest off-rate). For the tapasin deficient .220 cell line, the model predictions differ more substantially from the experimental data. In contrast to the .220tpn cells, in which presentation correlated with MHC-I affinity, the hierarchy of peptides in the .220 cell line is not solely determined by affinity for MHC-I. The extended *Dalchau* model provides some explanation for this phenomenon, with the observed greater presentation of SIINFEKM than the more stable SIINFEKV predicted by the model due to a higher TAP binding affinity. However, the model predicts SIINFEKL to be the most abundantly presented peptide in the .220 cell line, whereas the experimentalists found SIINFEKM to have the highest presentation.

Overall, the model predictions and experimental observations align closely, with the only prominent exception being the over prediction of SIINFEKL presentation in the tapasin-deficient .220 cell line.

5.3.2 Tapasin dependence in the extended Dalchau model

We investigated the effect of changing different parameters on the predicted tapasin dependence in the extended *Dalchau* model. 7 parameters were selected due to their predicted effect on tapasin dependence and were varied between 2.5 orders of magnitude above and below the maximum a posteriori parameter values found during the parametrisation of the extended *Dalchau* model. The perturbation of parameters directly corresponding to the effect of tapasin on the peptide loading process (e.g. the enhancement to peptide-MHC off rates, q) appeared unable to increase tapasin dependence by improving peptide presentation in presence of tapasin. This suggests that the MAP parameter values are already close to optimal for the system. Increasing the supply of tapasin in the ER leads to an increase in tapasin dependence but this increase plateaus at a tapasin dependence of only 2.60, indicating that high tapasin dependencies can only be obtained through low presentation in the absence of tapasin.

Reducing the peptide-MHC binding rate and increasing the peptide-MHC off-rate of the non-binding endogenous cohort both increase tapasin dependence by reducing presentation in the absence of tapasin. However, tapasin dependence plateaus at just above 100 when increasing u_n, which is far lower than the reported tapasin dependence of HLA-B*44:03, for example. We found that only by changing the peptide-MHC binding rate, b, could we obtain the full range of tapasin dependencies

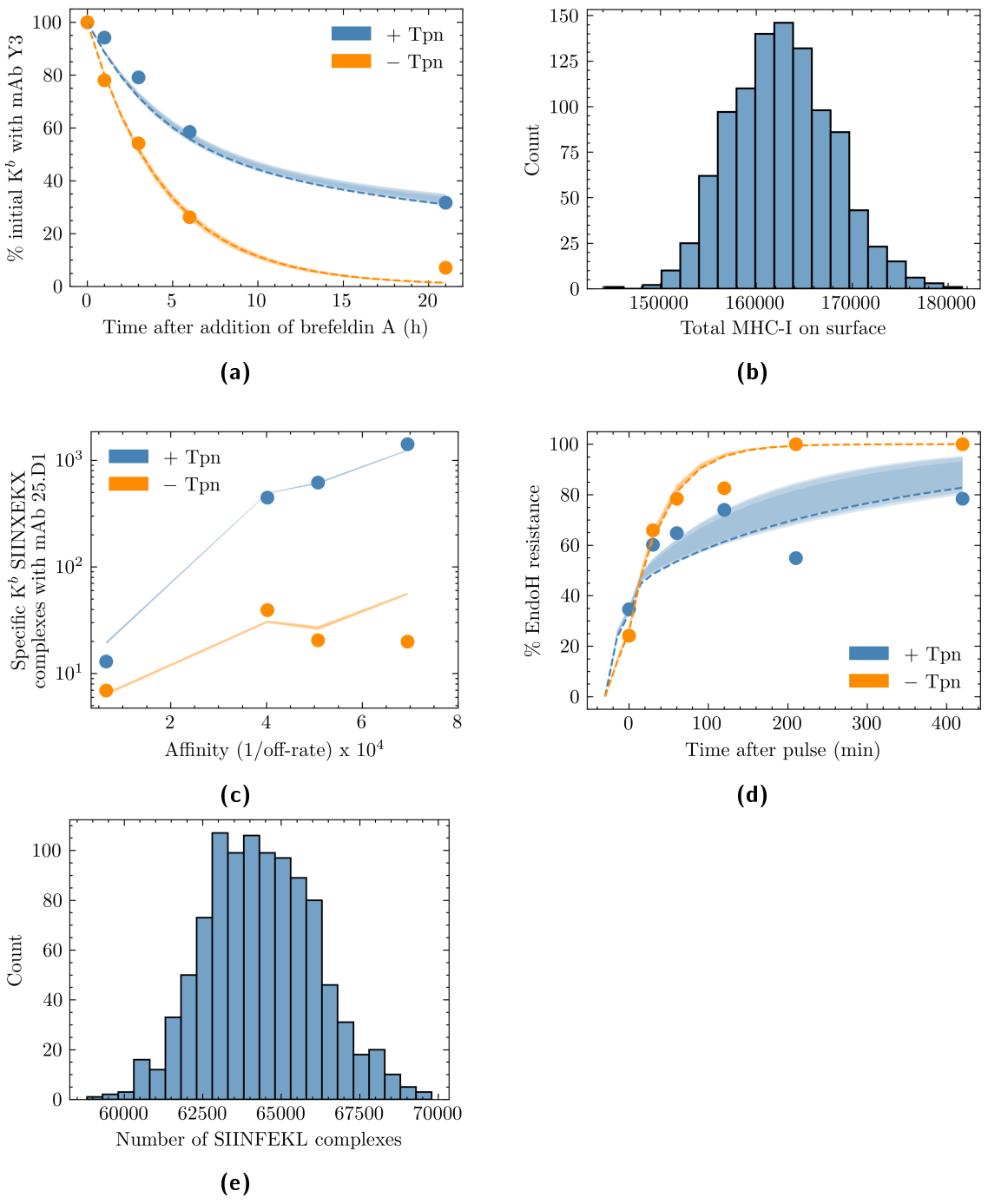


Figure 5.6. Comparison of model predictions (denoted by scatter points, where relevant) and experimental data. 1,000 sets of parameters were randomly sampled from the posterior distributions and used to simulate the experiments. Dashed lines indicate the predictions using the maximum a posteriori (MAP) parameters. Shaded regions indicate 97.5%, 90% and 50% credible intervals in order of increasing opacity.

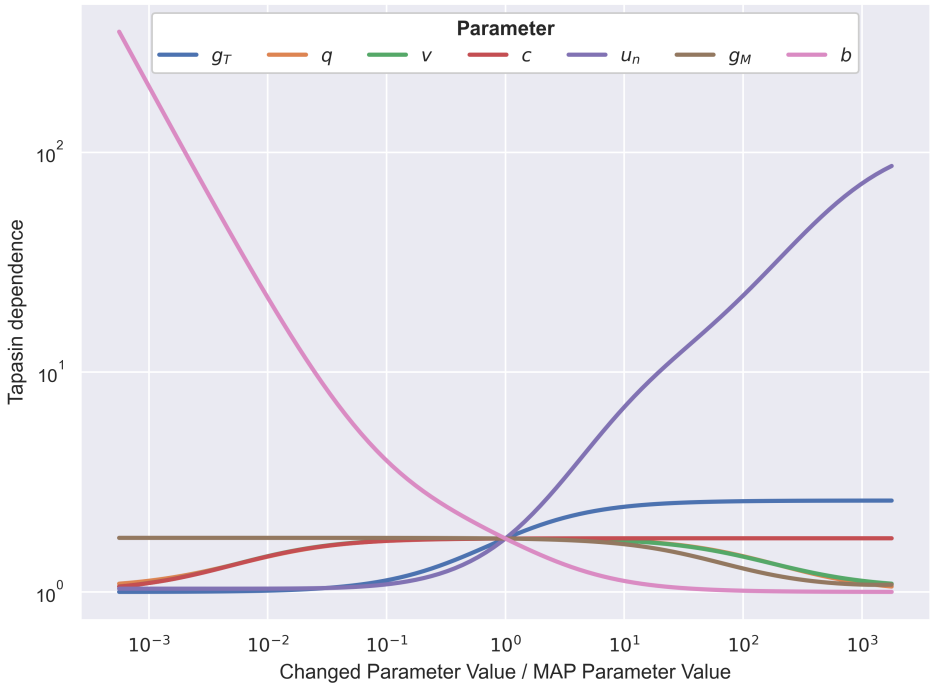


Figure 5.7. The predicted effect of changing parameters from the MAP values in the extended *Dalchau* model on the tapasin dependence.

reported by Bashirova et al., and hence concluded that this parameter should be used to tune tapasin dependence in the model.

This is consistent with the study of Bailey et al. into the structural intermediates of peptide loading using an extended version of the *Dalchau* model (mentioned in Section 5.2.1.1) [12]. They found that tapasin independent alleles moved from an open to a closed conformation at a faster rate than dependent alleles. Our model does not include multiple conformations of MHC-I, so the binding rate, b, can be thought of as encapsulating the binding of peptide to MHC-I and the conformational change of the MHC-I from an open to closed state. Hence, a higher value of b is consistent with a faster change in conformation, and hence a less tapasin dependent allele.

5.3.3 Prediction of tapasin dependence

5.3.3.1 Comparison of encodings and MHC-I representations

We tuned hyperparameters for support vector regressor models to predict the log-scaled tapasin dependence measurements reported by Bashirova et al. and compared trained model perforamance using a 10-fold cross-validation (Figure 5.8). For all

encoding strategies, we found that restriction of the MHC-I sequence to the F-pocket had a detrimental effect on predictive performance when compared to the equivalent model performance using the whole sequence or *NetMHCpan* pseudosequence. This suggests that tapasin dependence is not determined by the binding of tapasin at the F pocket, but by a property of the whole MHC-I (e.g. protein dynamics [12]).

Our attempts to include structural information in the model training had mixed success. Prediction of MHC-I structure using *AlphaFold2*, followed by structural encoding through *Foldseek* resulted in a worse-performing predictive model than using any of the standard amino acid encodings for all 3 representation strategies. However, using the pre-trained *ESM-2* to produce a latent representation of the MHC-I enhanced performance when using the full sequence. The mean MSE across the 10 folds was 0.138 using *ESM-2* — significantly lower than the Benner6 substitution matrix, which had the next lowest mean MSE of 0.160 (significant at the $p < 0.05$ threshold using a paired Wilcoxon test).

For the pseudosequence, the *ESM-2* encoding resulted in a higher mean MSE (0.186) than all of the standard amino acid encodings. This is likely because the pseudosequence does not correspond to a naturally occurring consecutive sequence of amino acids in the MHC-I structure. Thus the *ESM-2* Transformer encoder, trained on physical protein sequences, cannot produce a sensible latent representation.

5.3.3.2 Correlation of tapasin dependence and prediction error

The prediction error is dominated by a small subset of 9 HLA alleles, all with a mean MSE of over 0.5. 3 of these alleles belong to the HLA-B*44 supertype (B*44:02, B*44:03 and B*44:05) and are commonly studied for having significantly different tapasin dependencies despite close sequence similarity. The tapasin dependencies of HLA-B*44:03 and HLA-B*44:05 were measured by Bashirova et al. at 274.16 and 1.82 respectively. The other 6 poorly predicted alleles are all characterised by their high tapasin dependencies and, besides HLA-A*01:01 and HLA-B*49:01 (which were predicted with a mean MSE of only 0.045 and 0.363 respectively), were the other alleles with the highest tapasin dependencies in the training set. Hence, it appears that the allele prediction error correlates positively with its tapasin dependence.

5.3.3.3 Performance of best model

A scatter plot of the best performing method (full MHC-I sequence with *ESM-2* encoding) is shown in Figure 5.9. This plot shows that most error in the fitting process is the result of the under-prediction of highly tapasin dependent alleles.

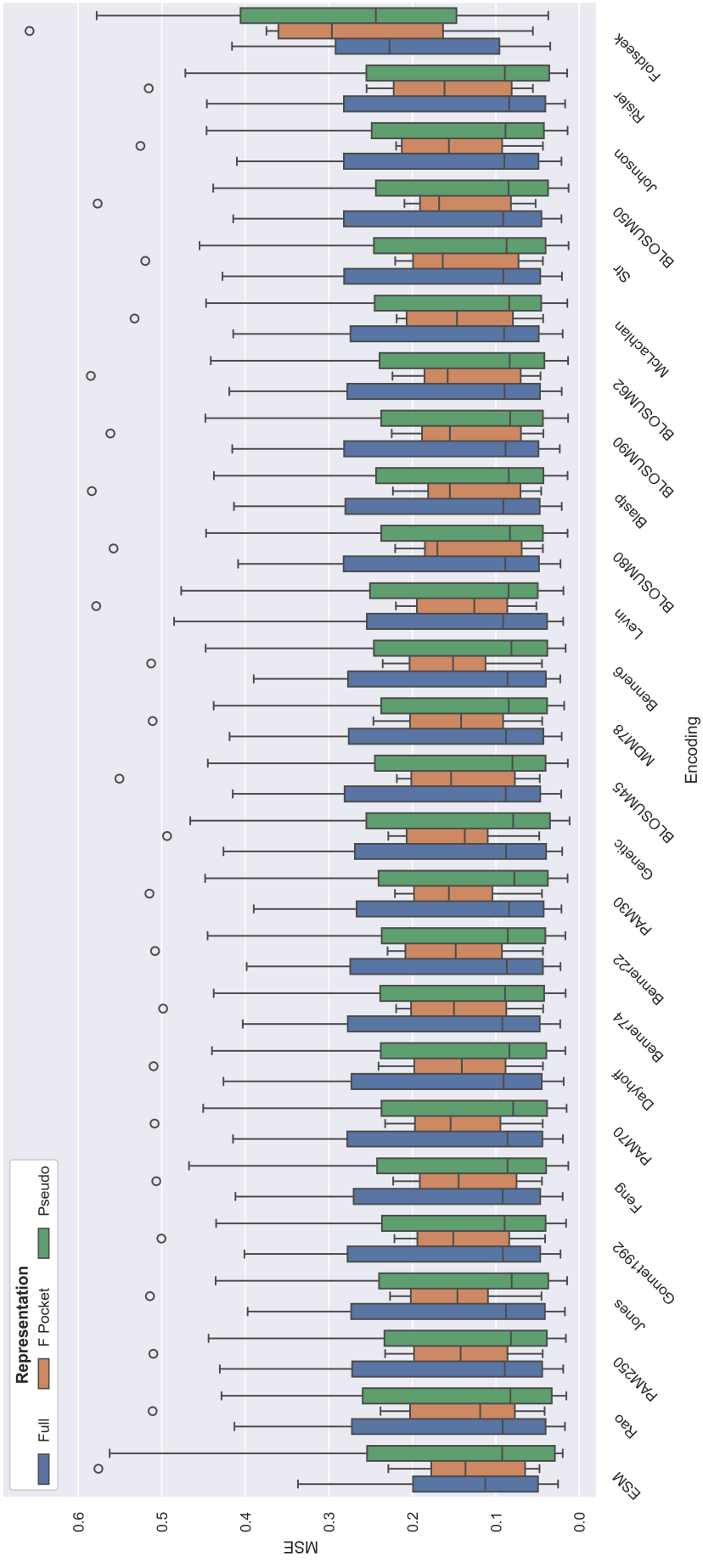


Figure 5.8. 10-fold cross validation performance of SVR model by MHC-I representations and different amino acid encoding strategies used. Model performance is shown as the mean squared error between predicted log-scaled tapasin dependence and Bashirova et al. measured tapasin dependence. Encodings are ordered by mean MSE, from lowest (left) to highest (right).

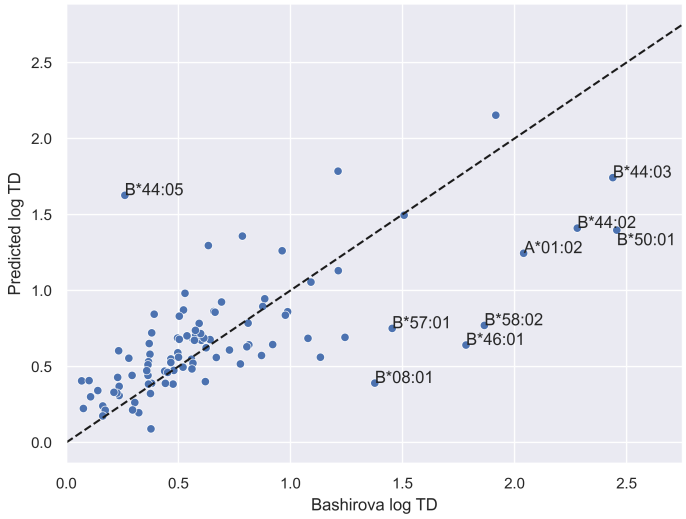


Figure 5.9. Scatter plot showing 10-fold cross-validation predictions for SVR model trained using *ESM-2* encoding of the full MHC-I sequence. Top 9 most inaccurately predicted alleles are annotated.

5.4 Discussion

5.4.1 Low SIINFEKL presentation in .220 cell line

We found the biggest discrepancy between predictions of the extended *Dalchau* model and the experimental data to occur in prediction of the presentation of SIINXEKX peptides in the tapasin deficient .220 cell line. In the standard *Dalchau* model, this manifested as an under-prediction of SIINFEKM presentation and was suggested to be caused by enhanced translocation of SIINFEKM by TAP. By integrating TAP translocation into the extended model, we successfully rationalised increased presentation of SIINFEKM relative to SIINFEKV due to the predicted TAP binding affinities of 45.0μM and 77.7μM, respectively, resulting in an amplified translocation rate of SIINFEKM, as Dalchau et al. had suggested. However, our model failed to predict the relatively low presentation of SIINFEKL (with a predicted TAP binding affinity of 35.6μM) compared to SIINFEKM.

All peptides were predicted to be trimmed slowly by ERAP1 and with only minor differences in efficiency. SIINFEKL was predicted to be trimmed with a catalytic rate of 0.167 s^{-1} — only fractionally higher than SIINFEKM (0.162 s^{-1}) or SIINYEKL (0.114 s^{-1}), and lower than SIINFEKV (0.176 s^{-1}). Hence, the inclusion of ERAP1 kinetics could not shed any light on the low SIINFEKL levels.

Assuming no significant experimental measurement error, it can therefore be inferred

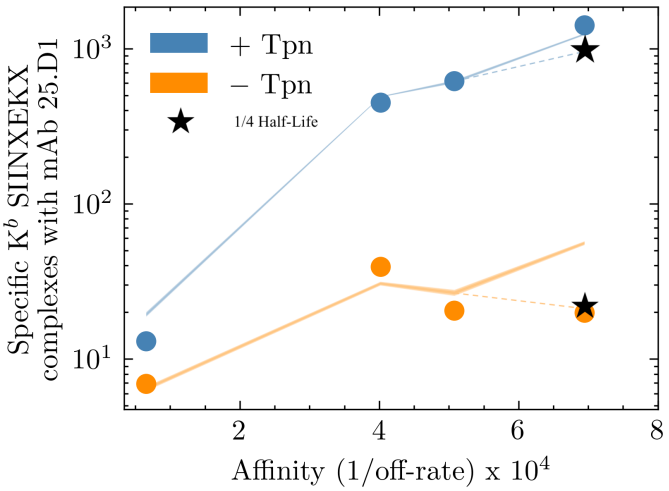


Figure 5.10. Dalchau SIINXEKX study simulated with SIINFEKL half-life in cytosol set to 25% of maximum a posteriori (MAP) value. The new predictions are denoted by a black star.

that an additional element within the antigen processing pathway, not encompassed by the extended *Dalchau* model, likely contributes to the diminished presentation of SIINFEKL. A prospective candidate for this role is ERAP2. Characterized by its preference for peptides of fewer than 9 amino acids, ERAP2 is anticipated to efficiently process all peptides. However, in contrast to ERAP1, ERAP2 does not exhibit binding affinity to the C-terminus of its substrate. Consequently, a notable discrepancy in peptide trimming between two peptides differing solely by a single amino acid at the C-terminus would not be anticipated. Nonetheless, the precise specificity of ERAP2 remains under thorough investigation in existing literature. Hence, it is plausible that ERAP2 may enhance the trimming rate of SIINFEKL relative to other peptides for reasons presently unknown. In particular, it should be noted that mice do not express ERAP2 so, although SIINFEKL is a canonical murine epitope, it is possible that a high level of destruction occurs when the epitope is introduced to the .220 human B-cell line.

Alternatively, it is possible that SIINFEKL might have a shorter half-life in the cytosol than SIINFEKM. Cytosolic half-life has been observed to vary across many orders of magnitude and appears to depend on the entirety of the peptide sequence. To test this hypothesis, we re-simulated the experiments with a cytosolic half-life of for SIINFEKL that was 1/4 that of the other 3 peptides. We found much closer agreement with the observed data (albeit with a slightly negative impact on the .220tpn prediction), indicating that this could be a possible explanation for the discrepancy with our original predictions (shown in Figure 5.10).

5.4.2 MHC-I on rate determines tapasin dependence in the extended Dalchau model

By varying different parameters in the extended *Dalchau* model, we found that the peptide-MHC binding rate was the only parameter capable of tuning tapasin dependence between the full range of values reported by Bashirova et al. in their study [14].

Intuitively, this means that tapasin independent alleles are characterised by efficient MHC-I binding without the assistance of tapasin, so logically the benefit of tapasin inclusion is reduced when compared with an allele for which MHC-I binding is much slower.

Physically, a higher binding rate can be thought of as the peptide-MHC complex moving from the *open* to *closed* conformation at a faster rate. This is then consistent with the observation that tapasin independent alleles, such as HLA-B*44:05, move to a closed state faster than tapasin dependent alleles [12].

5.4.3 Tapasin dependence cannot be accurately predicted by MHC-I sequence

Our attempts to predict tapasin dependence from MHC-I sequence were unsuccessful for highly tapasin dependent alleles, as shown by Figure 5.9. This was an anticipated outcome because alleles differing by only a single amino acid (e.g. B*44:02 and B*44:05) can have tapasin dependencies differing by almost 2 orders of magnitude, suggesting that MHC-I sequence similarity does not imply tapasin dependence similarity.

We see from Bailey et al.'s molecular dynamics simulations that this single amino acid substitution can cause changes in the plasticity of the tertiary structure that can explain the differences in tapasin dependence. More recent investigation of the protein dynamics of alleles with low tapasin dependency (Turner, Essex and Elliott, unpublished) has shown that these molecules edit peptide cargo by utilising an intramolecular hydrogen-bonding relay that is not deployed for tapasin dependent peptide editing.

Since prediction of protein structure from sequence remains an unsolved problem in the field (at least at the multiple conformation level), it is unsurprising that a model

trained only on protein sequence is unable to accurately predict a property that so closely corresponds to structure.

5.4.4 Single structure predictions do not improve model performance

We attempted to include structural information by predicting the structures of the 97 different HLA alleles using *AlphaFold2* followed by *Foldseek* to replace the MHC-I amino acid sequence with a sequence of structural tokens. However, performance using this method was significantly worse than using the MHC-I sequence with standard substitution matrices (Figure 5.8). This failure could be caused by either:

1. Inaccurate structure prediction of MHC-I by *AlphaFold*.
2. Inaccurate structure representation using *Foldseek*.

AlphaFold2 has been shown to accurately predict certain protein structures and has received a lot of attention for this accomplishment. However, *AlphaFold* has only solved structural prediction in a single structure universe [91]. Substantial research effort is currently being invested into generating ensembles of protein structures, either by tweaking *AlphaFold*'s multiple sequence alignment stage or through the development of novel predictive models, but for now we can only derive single structures from *AlphaFold*.

MHC-I tapasin dependence has been shown to correlate inversely with predicted protein plasticity using molecular dynamics simulations of B*44:02 and B*44:05 in their ligand-free states [12]. The B*44:02 molecule populated a single F-pocket conformation, whereas the tapasin independent B*44:05 allele populated several F-pocket conformations. Figure 5.11 shows the two structures predicted by *AlphaFold2* following structural alignment. The two structures are extremely closely similar with an RMSD of 0.22 Å. Although the two alleles differ by a single residue at site 116, even this region of the two predicted structures shows close alignment, despite being responsible for the two alleles occupying opposite ends of the tapasin dependence spectrum. The other conformations of B*44:05 are unknown because *AlphaFold2* has only returned a single structure. Hence, it is unsurprising that using these structures in our downstream prediction task led to to similar predictions of tapasin dependence for both B*44:02 and B*44:05.

AlphaFold2 returns the same structures because the ligand-bound structure of HLA-B*44:02 and HLA-B*44:05 are identical, as seen through alignment of the structures

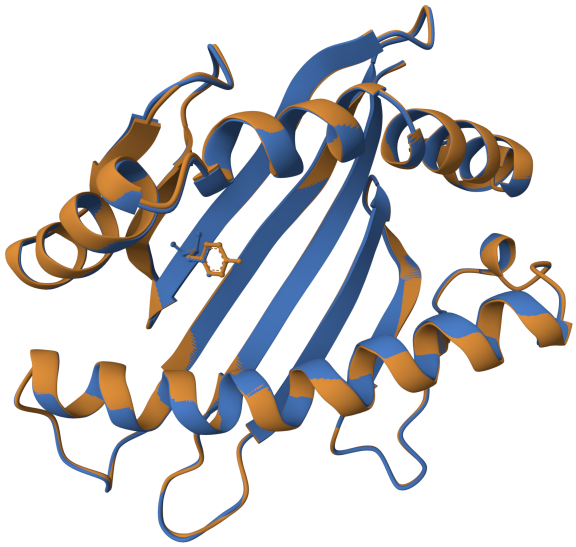


Figure 5.11. Structural alignment of the predicted structures of HLA-B*44:02 (orange) and HLA-B*44:05 (chain) using the jFATCAT algorithm through the RCSB web server. The two aligned molecules have a root mean squared distance of 0.22, a TM-score of 1.00 and a sequence identity of over 99% (single amino acid difference). The single residue difference at site 116 is highlighted using a ball-and-stick representation.

3L3K and 3KPP on PDB [107, 149]. As a conformational intermediate, the peptide free-structure would be expected to exist in a higher energy state, so we would not expect it to be returned by *AlphaFold2* (since the model has been trained on low energy states from PDB). As protein structural ensemble prediction continues to improve, it may soon be possible to predict these alternative conformations using machine learning. This could enable us to estimate protein plasticity without the need for molecular dynamics, thus facilitating the fast prediction of tapasin dependence for alleles not in the Bashirova data set.

5.4.5 Concluding remarks

In this chapter we have parametrised an extended version of the *Dalchau* model the addition of TAP and ERAP1, and have added conditions to ensure prediction of physically plausible numbers of pMHC complexes. We have shown how although the model is trained on H2-Kb it might be applied to any potential MHC-I allele by treating the peptide-MHC binding rate, b, as an allele-specific parameter.

This parameter can be estimated if the tapasin dependence is known. Although 97 measured levels of tapasin dependence have been reported in the literature, this only represents a small subset of the 26,610 HLA Class I alleles included in the

IPD-IMGT/HLA Database as of December 2023 [13]. Furthermore, the Bashirova dataset does not include any non-human MHC-I alleles (e.g. H2-Kb, which was used to parametrise the *DaIchau* model).

We therefore investigated how we might predict the tapasin dependence of alleles for which this figure has not been reported. We found that attempts to train a regressor to predict this quantity were largely unsuccessful. The best performance came from encoding the full MHC-I sequence using the protein language model, *ESM-2*, suggesting that consideration of MHC-I structure is key in the prediction of this quantity. However, we found that attempts to use the predicted structures from *AlphaFold2* did not enhance prediction, likely due to the fact that single structures only represent a snapshot of the protein's structure and do not permit the inference of protein plasticity — a key property in determining tapasin dependence.

Hence, accurate tapasin dependence prediction will require either the prediction of conformational ensembles of MHC-I structures or the dynamic simulation of protein structure through molecular dynamics simulations. The former is an area of active research currently and is computationally preferable due to the high computational cost of molecular dynamics simulation. As the next generation of protein structure prediction algorithms emerges in the wake of *AlphaFold2*, we anticipate that this might facilitate the accurate prediction of tapasin dependence for a far wider range of MHC-I alleles.

Chapter 6

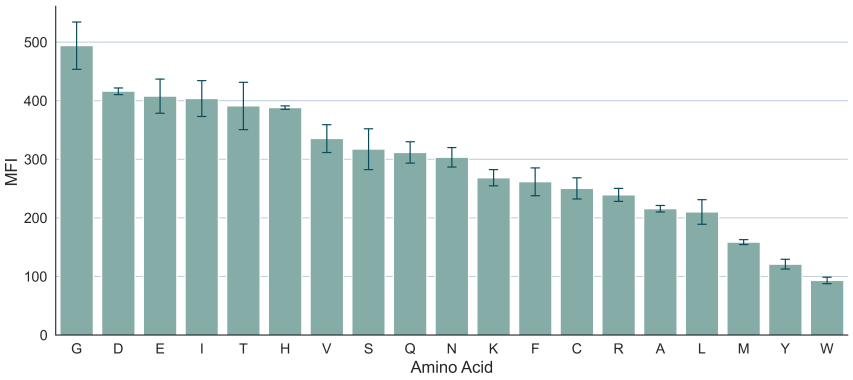
A mechanistic model of the antigen processing pathway

6.1 Introduction

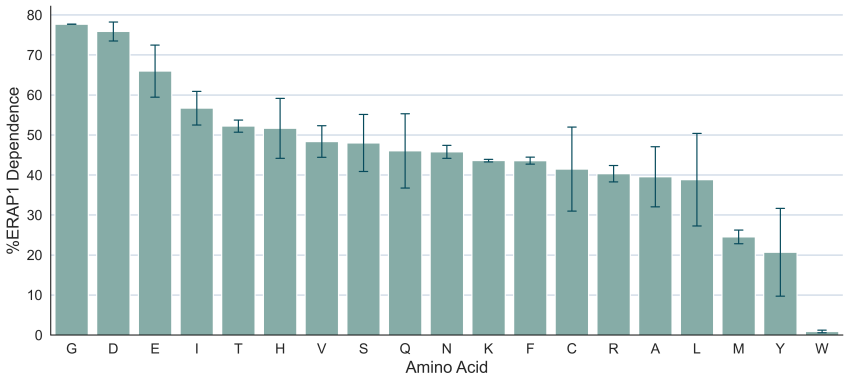
In Chapter 5, we extended an existing systems biology model of antigen processing to include the effects of TAP and ERAP1. Our analysis was limited by the data set used to re-parametrise the extended model. The data used had been collected by Dalchau et al. to facilitate the study of the MHC-I loading pathways in the endoplasmic reticulum [43]. The comparison of the tapasin deficient .220 cell line with the .220tpn transfected cell line permitted the parameters associated with the tapasin-dependent or independent peptide loading pathways to be identified (shown by tight posterior distributions in Figure 5.5), but not the effects of TAP or ERAP1. We also neglected to consider the effects of cytosolic aminopeptidases under the assumption that none of the associated parameters would be identifiable from the data, and because all 4 SIINXEKX peptides had identical N-termini so would likely have similar aminopeptidase activity.

In this chapter, we use a more comprehensive study of the effect of precursor N-terminus on antigen processing efficiency to determine the remaining parameters in our mechanistic model and validate its efficacy on a larger range of peptides [71]. We use data from two experimental assays reported in this study:

1. Antigen processing efficiency of ubiquitinated N-terminally extended SIINFEKL peptides (ub-XXSL) in wild type (WT) HeLa cells (Figure 6.1a).



(a)



(b)

Figure 6.1. (a) Mean fluorescence intensity (MFI) from specific binding of 25.D1 antibody to H2-Kb bound SIINFEKL after 48h following transfection with ubiquitinated XXSIINFEKL (ub-XXSL), where X denotes one of 19 different amino acids. (b) Dependence on ERAP1 calculated by treatment of HeLa cells with siRNA against ERAP1 and evaluating difference against wild type. Error bars represent the results of two independent experiments. All data are reproduced from [71]

2. Antigen processing efficiency of ubiquitinated N-terminally extended SIINFEKL peptides (ub-XXSL) in HeLa cells treated with ERAP1-specific siRNA (Figure 6.1b).

Because of the diversity of N-termini across the XXSL peptides, we anticipated that cytosolic aminopeptidases would likely be partially responsible, along with ERAP1, for the trimming of these precursors in an N-terminus dependent manner [6, 139]. We therefore decided to include the data from the ERAP1-deficient siRNA treated HeLa cell line, hypothesising that this would enable us to separate peptide trimming in the cytosol from peptide trimming in the ER.

Hence, in this chapter we are able to fill in the gaps in our model of the antigen processing pathway remaining from the extended *Dalchau* model presented in Chapter

5. This leaves us with a fully parametrised model, theoretically capable of accurately predicting raw numbers of pMHC complexes on the cell surface. We finish the chapter with a sensitivity analysis of this model, along with a discussion of its limitations.

6.2 Methods

6.2.1 Mechanistic model fitting

6.2.1.1 Hearn data set

To validate our mechanistic model and infer the role of cytosolic aminopeptidases, we utilised a study by Hearn et al. into the effect of the N-terminus extension on the efficiency of generation and presentation of SIINFEKL [71]. The authors constructed plasmids consisting of ubiquitin (Ub) at the N-terminus of XXSIINFEKL (XXS-L), where X represents one of 19 amino acids (all canonical amino acids, excluding proline). H2-Kb transfected HeLa cell lines were then transiently transfected with the plasmid and incubated for 24-48 h, at which point H-2Kb-SIINFEKL complexes were quantified on the cell surface using the 25.D1.16 monoclonal antibody (mAb).

Hearn et al. then repeated this study using a HeLa cell line that had been treated with small interfering RNA (siRNA) for ERAP1, knocking down levels of the enzyme to below 10% of WT levels. They used the difference in presentation of SIINFEKL in this ERAP1 knockdown cell line to calculate the dependence of the different XXS-L peptides on ERAP1 for the generation of the epitope according to the formula:

$$\%\ \text{ERAP1 dependence} = 100\% - \frac{\text{ERAP1 knockdown SIINFEKL}}{\text{WT SIINFEKL}}. \tag{6.1}$$

All data were extracted from the original figures using the Web Plot Digitizer tool [132].

6.2.1.2 Cytosolic aminopeptidases

We adapted the extended *Dalchau* model presented in Chapter 5 to incorporate substrate-specific N-terminus trimming by cytosolic aminopeptidases. Many different cytosolic aminopeptidases have been identified and are implicated in epitope production, including puromycin-sensitive aminopeptidase (PSA), leucine aminopeptidase

(LAP), and bleomycin hydrolase (BH). These aminopeptidases have been noted to have considerable redundancy in their specificities, and there may still aminopeptidases which are yet to be identified in the cytosol. It would therefore be highly complicated for modelling purposes to consider these aminopeptidases as independent enzymes. Instead, we made the parsimonious decision to treat them as a single collective species of cytosolic aminopeptidases, the specificity of which we attempted to infer from trimming in the ERAP1-depleted HeLa cell line.

We modelled the rate of substrate trimming by cytosolic aminopeptidases through a simple trimming rate, r_X (where X denotes the N-terminus amino acid), under the assumption that this rate depends only on the N-terminus of the substrate. This term was added to the equations describing free peptide in the cytosol in the extended *Dalchau* model, resulting in a new system of ordinary differential equations given in Section A.2.

6.2.1.3 Off-rate measurement

Previous analysis had indicated that the *Dalchau* model was very sensitive to the peptide-MHC off-rate parameter. This can be seen through the differences in presentation of the 4 SIINXEKX peptides in the .220tpn cell line (shown in Figure 5.6e). Off-rates can be predicted by various publicly available algorithms, but this introduces high uncertainty into this parameter [114, 118]. To limit uncertainty and hence enhance our confidence in the identification of the remaining unknown parameters, we decided to experimentally measure the off-rates of the 19 XXS-L peptides, as well as the 19 intermediate products of N-terminus processing (XS-L), and the off-rate of SIINFEKL itself.

Peptides were ordered from GenScript, with purity of over 98%. Off-rates were measured using a brefeldin A decay assay with an RMA-S cell line. For each peptide, between 3 and 5 replicates were taken. An exponential decay was fitted to the resulting time series using Bayesian inference with MCMC to capture the noise observed through the experimental repeats. The maximum a posteriori off-rate values were chosen for use in further simulation. The lab reported problems in obtaining a high purity of IISIINFEKL, so calculation of the off-rate for this peptide was not possible.

6.2.1.4 Parameterisation

We fixed most parameters in the model in Section A.2 to the maximum a posteriori (MAP) parameter values determined through the inference of the extended *Dalchau*

model in Chapter 5, with only 4 exceptions. Of these, the peptide-MHC egress rate (e), peptide-TAP dissociation rate (u_S) and concentration of ERAP1 (E_0) were deemed to be non-identifiable from the data used to parametrise the model due to their broad posterior distributions. We also chose to refit the degradation rate of peptide in the cytosol (d_C) as we had chosen to separate cytosolic aminopeptidase activity from degradation by other resident peptidases. The off-rate of IISIINFEKL (u_{II}) required fitting as it was not possible to measure experimentally.

The addition of trimming by cytosolic aminopeptidases introduced a new parameter, r_X, to be determined for all 19 amino acids studied in the Hearn dataset. We assumed that the generation of each XXS-L peptide in the cytosol following the removal of ubiquitin was approximately homogeneous across the 19 amino acids, with rate g_i.

Hearn et al. measured surface pMHC levels using fluorescence due to binding of the 25.D1.16 mAb. We assumed that the observed fluorescence was linearly proportional to the level of pMHC on the cell surface, so could be related through a scaling factor. With the addition of this scaling factor, we were left with 26 parameters requiring estimation.

6.2.1.5 Model simulation

For each proposed set of values of the 26 unknown parameters, θ, we simulated the model described by the system of ordinary differential equations in Section A.2 for each of the 19 different XXS-L peptides. The system was initialised to its equilibrium point in the absence of exogenous peptide (i.e. $g_{XXSL} = 0$ in Equation A.22). The exogenous peptide supply term was then set to g_P and the system simulated until a time of 36 hours (chosen because the original paper only specified that cells were incubated for 24-48 hours). At this time, the level of SIINFEKL on the cell surface was taken and scaled by the scaling factor:

$$f_X^{WT} = \sigma \times M_e P_{SL}, \tag{6.2}$$

where X denotes the N-terminus extension amino acids, now removed from the SIINFEKL.

To simulate the ERAP1 knockdown cell line, we set ERAP1 levels to 10% of the levels in the wild type cell line and re-simulated the ODE system until a time of 48 hours (as the authors measured presentation for this cell line on day 3). At this time,

the level of SIINFEKL on the cell surface was taken and scaled, and used to calculate the ERAP1 dependence by:

$$f_X^{KO} = \left(1 - \frac{\sigma \times M_e P_{SL}}{f_X^{WT}}\right) \times 100\%.$$

6.2.1.6 Bayesian inference via MCMC

In order to detect any non-identifiable parameters, we used a Bayesian inference approach to sample the inferred posterior probability distributions of the 26 parameters using a Markov Chain Monte Carlo (MCMC) method. We normalised the observed fluorescence in the WT cells to be between 0 and 100 so that it was on the same scale as the observed ERAP1 dependence percentage.

Hearn et al. took 2 repeats for each measurement, giving some indication of the level of experimental noise. We decided that 2 points was not sufficient to determine the distribution of errors, so made the commonly used naive assumption that the observed data (y_X^{WT} and y_X^{KO}) were Gaussian distributed about the model predictions. We included both repeated measurements in the calculation of our log-likelihood function, given by:

$$\ell(\theta) = -2N\log(2\pi) - 4N\log(\sigma) - \frac{1}{2\sigma^2}\sum_{i=1}^{2}\sum_{j=1}^{N}\left[\left(y_{X_j,i}^{WT} - f_{X_j}^{WT}\right)^2 + \left(y_{X_j,i}^{KO} - f_{X_j}^{KO}\right)^2\right], \tag{6.3}$$

where $N = 19$ is the number of amino acids, the dummy index i is used to sum over both repeats, and the dummy index j is used to sum over the different amino acids, given by X_j.

We fixed the standard deviation of the noise, σ, to be 10 (i.e. approximately 10% of the maximum observation), although the set of parameter values, θ^{MLE}, minimising $\ell(\theta)$ is independent of the selected value of σ, so we did not place much importance on this parameter assignation.

Due to the high computational requirements of MCMC and the large number of parameters requiring estimation, we decided against using a uniform log prior (as we had done in Chapter 5). Instead, we ran an initial parameter optimisation, starting at a position chosen uniformly at random in 26 dimensional space between the assumed upper and lower bounds for each parameter. We maximised the Gaussian log-likelihood given in Equation 6.3 by using the CMA-ES algorithm through the optimisation toolkit in PINTS [40]. We then used this set of maximum likelihood

parameters, θ^{MLE}, as the mean of truncated Gaussian log prior distributions with standard deviation equal to the difference between the upper and lower bounds, divided by 8 (this was chosen as a compromise between computational efficiency and permitting the sampling of a broad range of parameter space).

We initialised 3 Markov chains at random states sampled from the prior and ran them using the Haario Bardenet ACMC algorithm in the PINTS toolkit. After 350,000 iterations, we confirmed that the 3 chains had converged to their posterior distributions by ensuring that the convergence criterion, $\hat{r}$, was less than 1.05 for all parameters, and by inspection of the trace plots.

6.2.2 Global sensitivity analysis

To determine the epitope- and precursor-specific parameters with the most influence on the presentation of the epitope, we conducted a global sensitivity analysis on the mechanistic model. We used extended Fourier amplitude sensitivity testing (eFAST). This method yields two indices for each parameter: (i) the first-order sensitivity index, S_i, denoting the proportion of the output's variance due only to the given parameter, and (ii) the total-order sensitivity index, S_{T_i}, capturing higher order effects due to interactions between the different parameters.

The first-order sensitivity index is calculated by varying each parameter at unique frequencies. The variance in the model output at the parameter's unique frequency then indicates the effect of the given parameter on the model output. The total-order sensitivity index is a result of an extension proposed by Saltelli and Bolado to the basic Fourier amplitude sensitivity testing, in which the complement set of parameters is varied at low, not necessarily unique frequencies while the parameter of interest is being varied at a high, unique frequency [135]. The sensitivity indices of the complementary set of parameters are summed to give their total contribution, S_{C_i}, and the remaining variance is then obtained as $S_{T_i} = 1 - S_{C_i}$.

We used the SALib package in Python to implement eFAST on our mechanistic model. We adapted the model used to simulate the Hearn study by considering the production of an 8mer epitope and its N-terminally extended precursors up to a length of 16 amino acids by the proteasome. For each set of parameter values, we simulated the model to equilibrium and recorded the presentation of the 8mer. In our analysis, we investigated the effect of 5 types of parameter on the presentation of the 8mer: (i) production rate by the proteasome (g_i), (ii) cytosolic aminopeptidase trimming rate (r_i), (iii) peptide–TAP binding affinity (u^S/b_i^S), (iv) ERAP1 trimming rate (k_i^{ER}), and (v) peptide–MHC-I off-rate (u_i). Thus, we considered 5 different

Parameter	Lower	Upper	Units
g_i	0.0	3.0	peptides s^{-1}
r_i	-5.0	-0.5	s^{-1}
u^S / b_i^S	0.0	8.0	nM
k_i^{ER}	-2.0	1.5	s^{-1}
u_i	-6.0	-3.0	s^{-1}

Table 6.1. Parameter symbols and ranges of $\log_{10}$ values used for eFAST sensitivity analysis.

parameters for each of the 9 peptide lengths, giving us 45 parameter values in total. We log-scaled parameter values and sampled them uniformly from the ranges shown in Table 6.1.

We also included a dummy parameter which had no effect on the mechanistic model output. This was considered to provide a baseline for statistical comparison, as has been proposed previously in the literature [108]. We conducted the sensitivity analysis 5 times (for random seeds = 0, 1, ... , 4) and conducted a one-tailed t-test to compare the sensitivity indices of each parameter with those of the dummy parameter and check for a significant increase. This gave us a p-value for each sensitivity index, indicating the probability that the increase in that parameter could be due to random chance.

For each parameter, 1,000 samples were generated during each running of the sensitivity analysis. Hence, we simulated the mechanistic model for 46,000 different combinations of parameter values for each of the 5 random seeds. We had originally assumed that the tapasin dependence of the restricting MHC-I allele would affect the sensitivity to different parameters. Therefore, we conducted the sensitivity analysis for 3 different hypothetical alleles, with low, medium and high tapasin dependencies of 1.25, 10 and 100 respectively. However, we found no notable differences between the 3, so only present our findings for the low tapasin dependence allele in the results in Section 6.3.3.

6.3 Results

6.3.1 Hearn fitting results

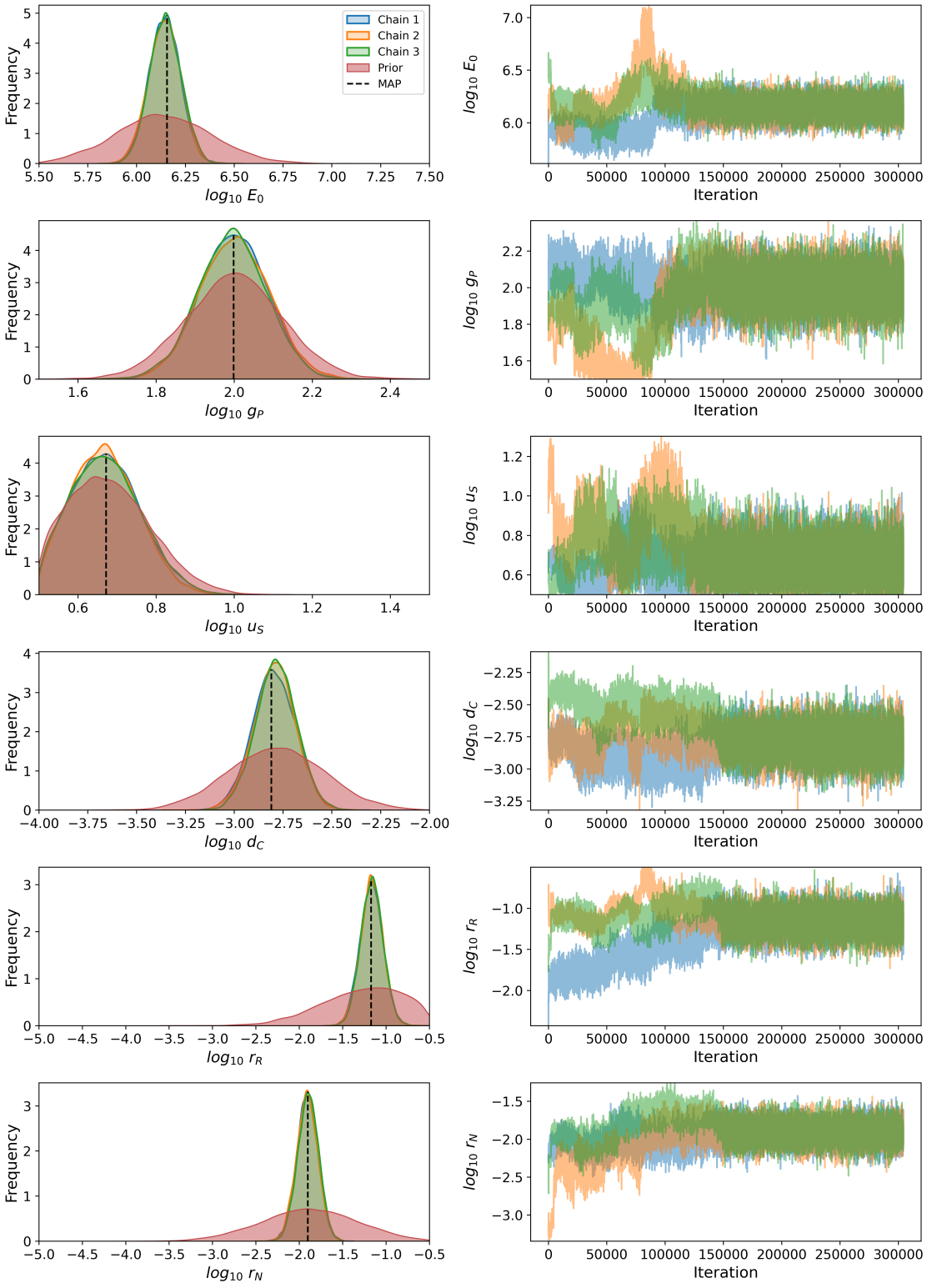


Figure 6.2

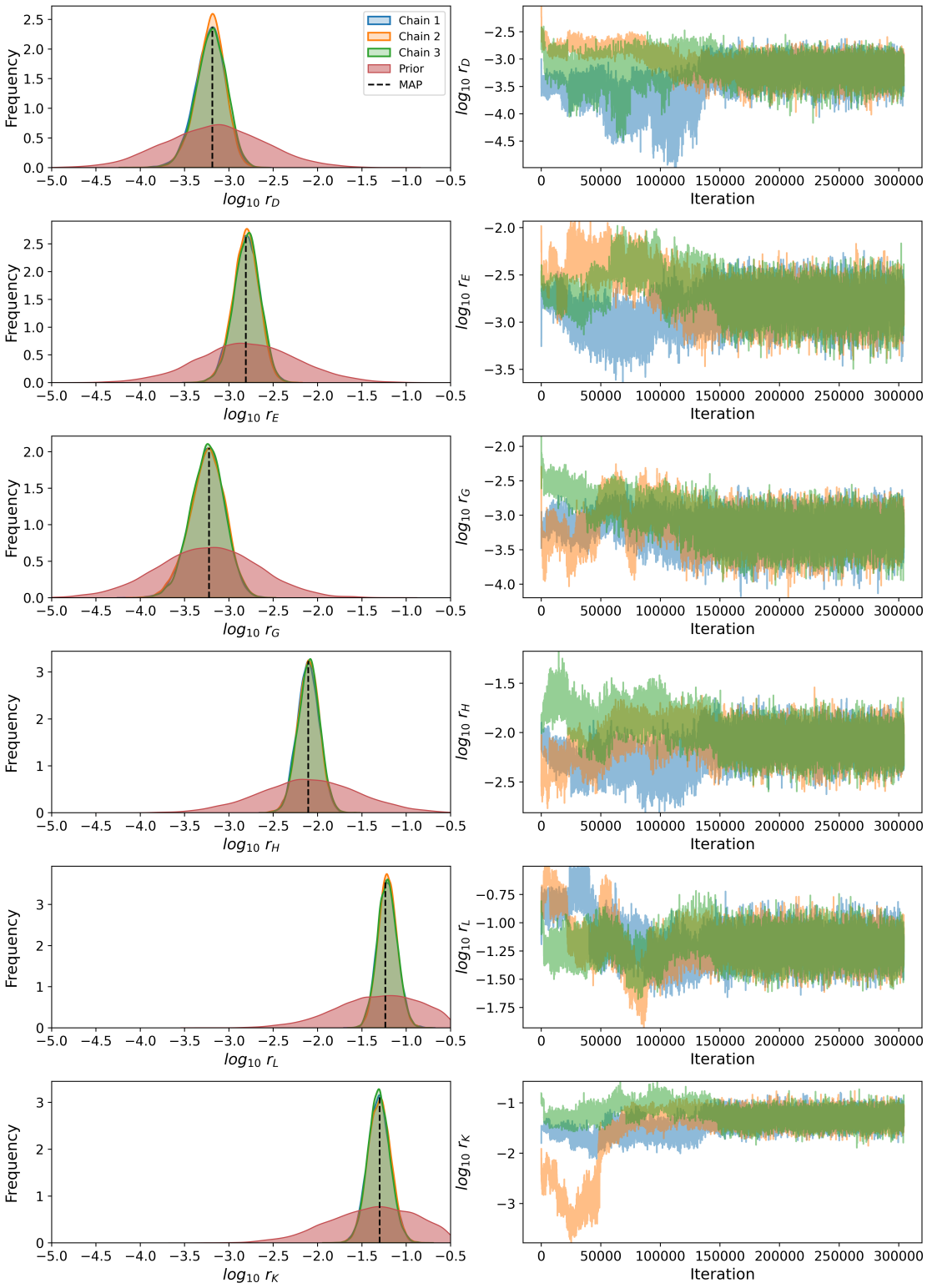

Chain 1
Chain 2
Chain 3
Prior
MAP
Frequency
$log_{10}\ r_D$
$log_{10}\ r_E$
$log_{10}\ r_G$
$log_{10}\ r_H$
$log_{10}\ r_L$
$log_{10}\ r_K$
Iteration

Figure 6.2

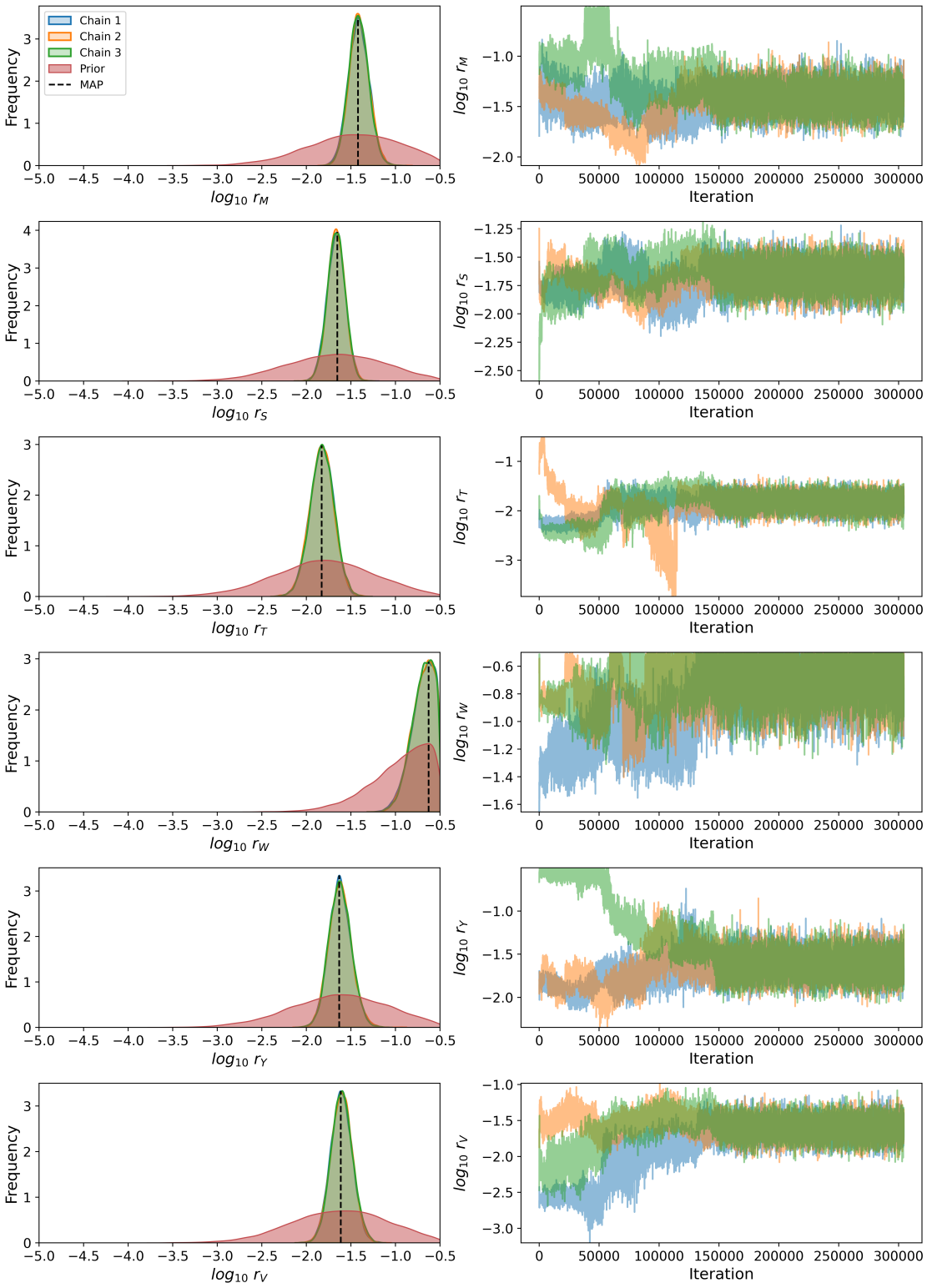

Chain 1
Chain 2
Chain 3
Prior
MAP
Frequency
$log_{10}\ r_M$
$log_{10}\ r_S$
$log_{10}\ r_T$
$log_{10}\ r_W$
$log_{10}\ r_Y$
$log_{10}\ r_V$
Iteration

Figure 6.2

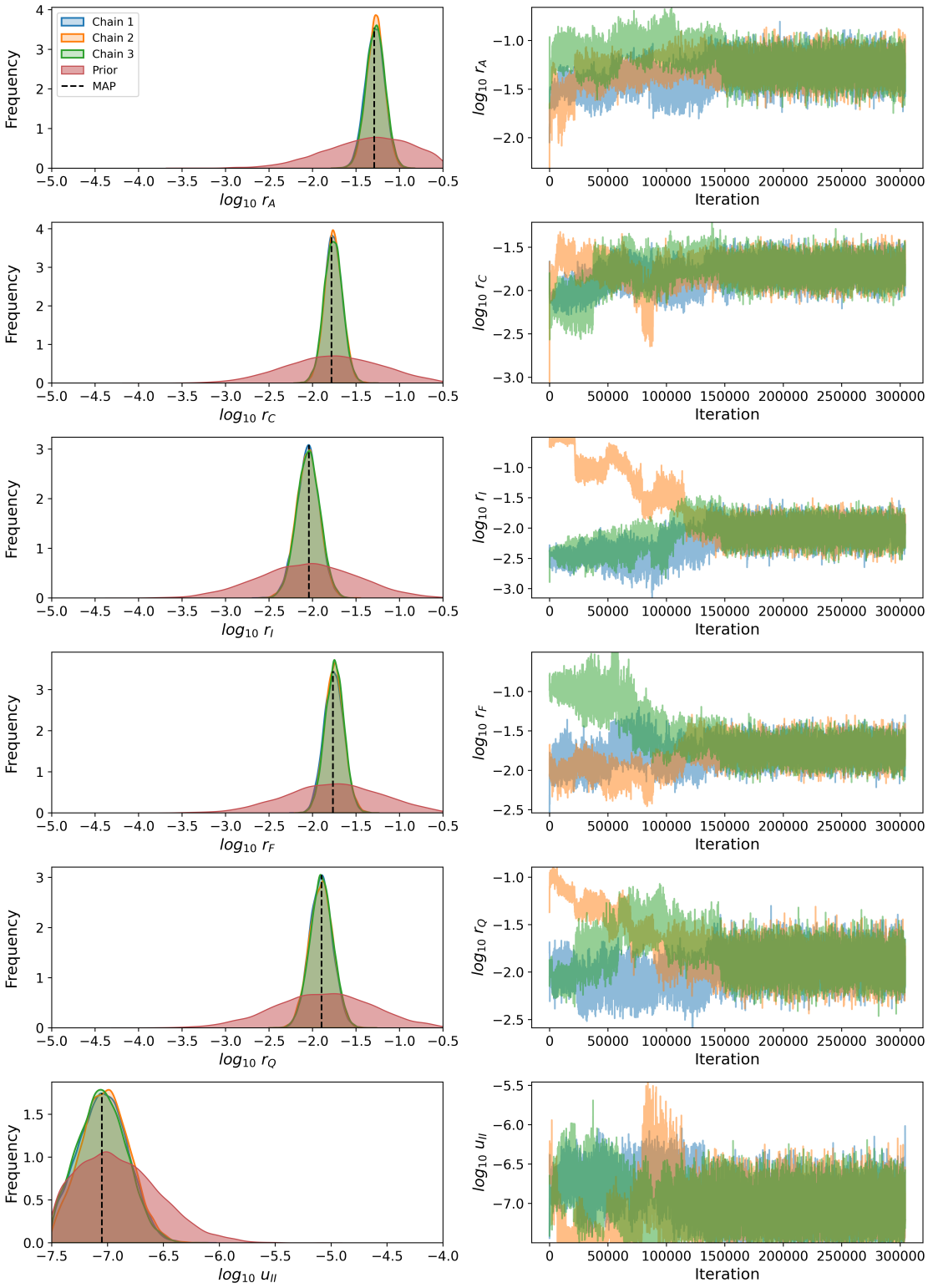

Chain 1
Chain 2
Chain 3
Prior
MAP
Frequency
$log_{10}\ r_A$
$log_{10}\ r_C$
$log_{10}\ r_I$
$log_{10}\ r_F$
$log_{10}\ r_Q$
$log_{10}\ u_{II}$
Iteration

Figure 6.2

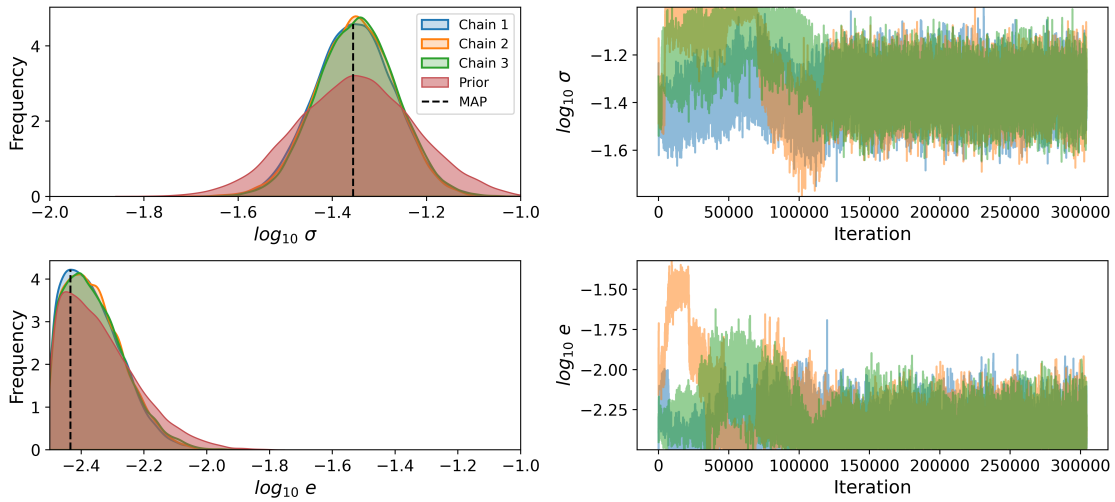
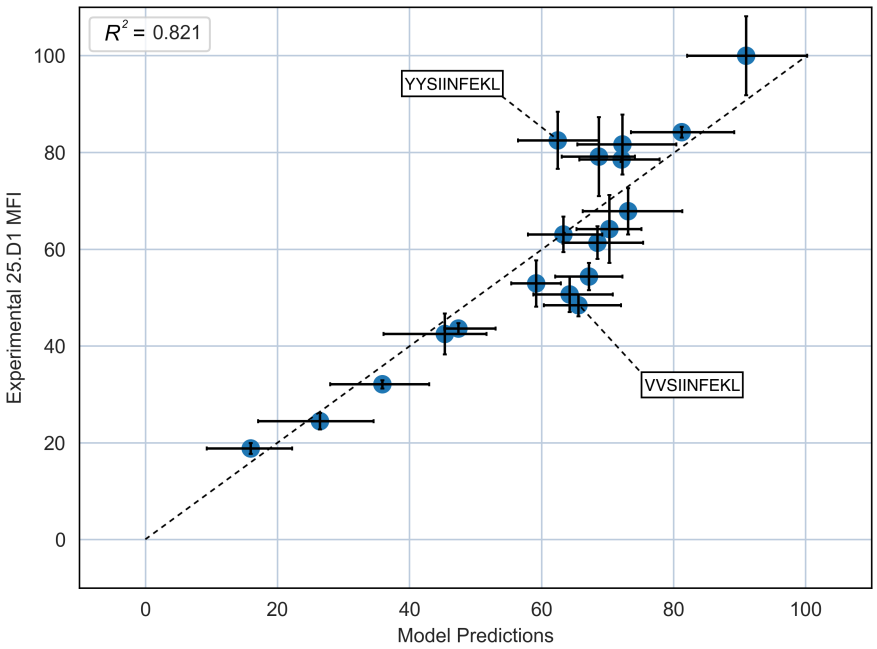


Figure 6.2. Posterior distributions (left) and trace plots (right) showing result of MCMC for the 26 inferred parameters. Posterior distributions were calculated from the 3 Markov chains after discarding the first 150,000 samples (the burn-in period, deduced from the trace plots) using the Seaborn library kernel density estimate (*kdeplot*) with default parameters. The maximum a posteriori (MAP) parameters are indicated by a dashed line for each posterior parameter distribution.

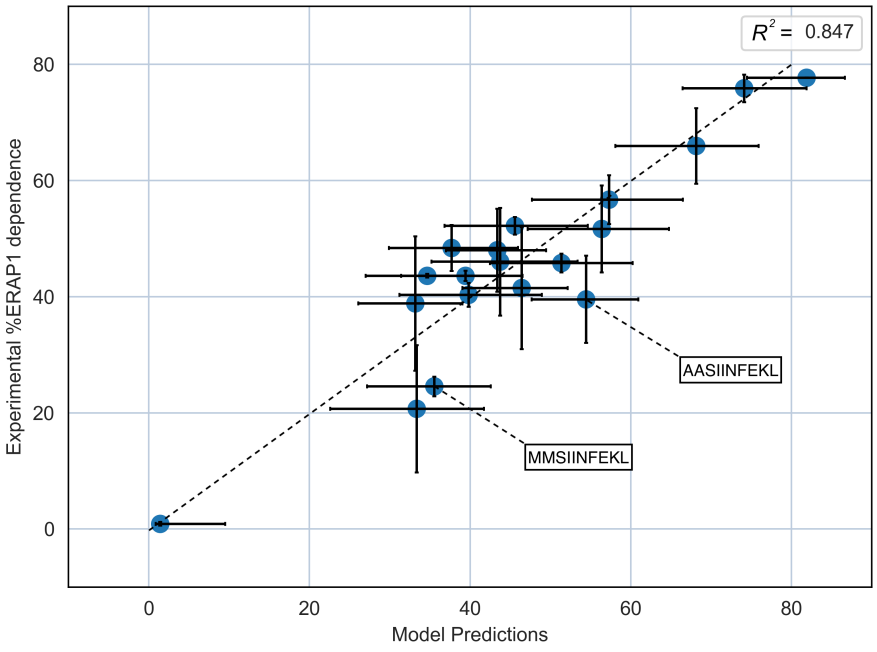
The results of the Bayesian inference of unknown parameters in the mechanistic model are shown in Figure 6.2. All three chains had converged after approximately 150,000 iterations. Kernel density estimate plots for the 150,000 iterations following this point show that the maximum a posteriori (MAP) parameter values are close to the maximum likelihood estimate (MLE) parameter values used as the mean of the prior distributions.

The comparison between the posterior distributions and the prior distributions suggests that most parameters were practically identifiable from the available data, since alternative parameter values in the local vicinity of parameter space were rejected during the sampling.

We sampled 1,000 sets of parameters from these posterior distributions and simulated the mechanistic model for each set. A comparison between the distribution of model predictions and the observed data by Hearn et al. is shown in Figure 6.3. A high degree of consistency between the mechanistic predictions and observations can be seen for both the WT presentation and the ERAP1 dependence. There are a couple of peptides for which we were not able to obtain consistency between our predictions and the observations. These are annotated in the plots to show the full XXSL sequence.



(a)



(b)

Figure 6.3. Comparison between mechanistic predictions using maximum a posteriori (MAP) parameters and the observations of Hearn et al. for (a) SIINFEKL presentation in WT HeLa cells, and (b) ERAP1 dependence in siRNA treated HeLa cells. Error bars for x-axis indicate 95% credible interval following sample of 1,000 parameters from posterior distributions. Error bars on y-axis indicate results of two independent experiments.

6.3.2 Comparison with Schatz study

Having obtained tight posterior distributions for the predicted trimming rates of 19 amino acids by cytosolic aminopeptidases, we wanted to investigate whether this was consistent with the limited experimental data in the literature studying the activity of this collection of enzymes.

We used a study by Schatz et al. in which the authors extracted and purified cytosol from human cell lines before studying the *in vitro* trimming of the 9mer XRGYVYQGL (where X denotes E, P, S, Q, I, R, L, F, M, or W) by the purified extract.

6.3.2.1 Schatz parameter inference

We fitted an exponential decay curve to the Schatz data, assuming that the trimming rate was dependent on the N-terminus of the substrate. Posterior distributions for the trimming rates of 10 peptides were sampled using a Bayesian inference approach with MCMC. 3 Markov chains were generated using the Haario-Bardenet algorithm through the PINTS toolkit [40]. We assumed multiplicative Gaussian noise for the log likelihood function so that error in the observed substrate concentration was proportional to the magnitude of the measurement. The summary statistics for the posterior distributions of the trimming rates are given in Table 6.2. The convergence criterion, $\hat{r}$, being less than 1.05 for all parameters indicates successful convergence of the 3 chains [33].

To investigate the consistency of the proposed model and resulting parameter distributions with the experimental data, we randomly sampled 1,000 sets of parameters from the posterior distributions. The plots in Figure 6.4 show close agreement between the 50%, 90%, and 95% credible intervals and the measurements of Schatz et al. across all 10 peptides. This demonstrates that the collective activities of the enzymes can be accurately represented by a single parameter.

6.3.2.2 Comparison of Hearn and Schatz parameters

We found a strong correlation between the posterior distributions of cytosolic aminopeptidase trimming rates determined in the parametrisation of our mechanistic model using Hearn et al.'s study and those determined using Schatz et al's *in vitro* degradation assay (shown in Figure 6.5). We fitted a linear regression between the median log-scaled mean parameter values from each method, determining Pearson and Spearman correlation coefficients of $R_p = 0.812$ and $R_s = 0.783$ respectively, and a line of best fit given by the equation $y = 1.16x + 2.60$.

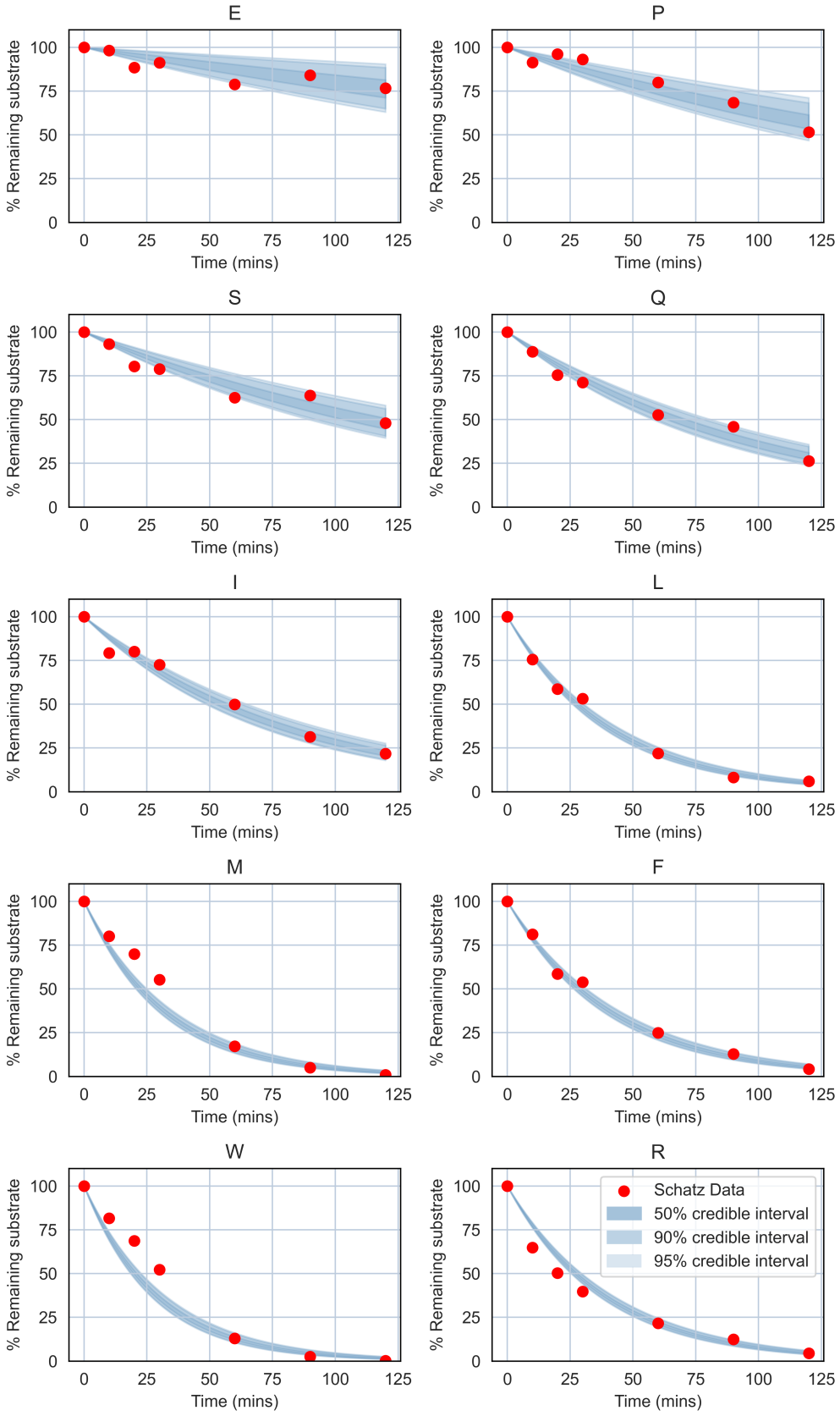


Figure 6.4. Results of fitting exponential decay to data from the *in vitro* trimming study of Schatz et al. [139] (shown by red points). 1,000 trimming rates for each peptide were sampled from the posterior distributions. Their credible intervals are denoted by blue shading.

N-terminus	Mean	Std.	ESS	$\hat{r}$
E	−4.45	0.17	8141.56	1.00
P	−4.12	0.08	8469.52	1.00
S	−3.99	0.06	8825.65	1.00
Q	−3.78	0.04	8515.18	1.00
I	−3.68	0.03	8250.84	1.00
L	−3.38	0.02	8396.68	1.00
M	−3.29	0.02	8644.78	1.00
F	−3.39	0.02	8628.94	1.00
W	−3.25	0.02	8564.35	1.00
R	−3.37	0.02	9000.31	1.00

Table 6.2. Summary statistics from Bayesian inference of trimming rates of peptides in the Schatz et al. data set. The mean and standard deviation (std.) of the posterior distribution is shown for each parameter. Effective sample size (ESS) and the convergence criterion, $\hat{r}$, are also provided to show convergence of chains.

The raw values of the rates determined in the Schatz study differ from those determined using the Hearn study. However, we assume that this is the result of the difference in enzyme concentration between the purified cytosolic extract and the *in vivo* cytosolic conditions in Hearn et al.'s HeLa cell line.

Hearn et al. did not study peptides with proline N-termini because of complications with the removal of ubiquitin when bound to proline *in vivo*. In order to therefore determine a corresponding trimming rate for proline to be used for our mechanistic model, we used the line of best fit to calculate the rate from the mean of the trimming rate determined through the fitting of the Schatz et al. data. This is denoted by a red point in Figure 6.5.

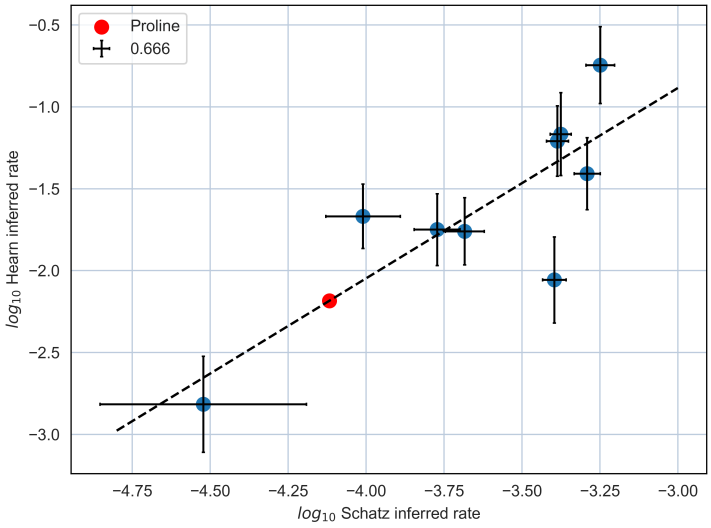


Figure 6.5. Comparison of trimming rates resulting from the Schatz study inference (shown on the x-axis) and the inference using the Hearn study (y-axis). Error bars span the 95% credible interval from the posterior distributions of the Bayesian inferences. The line of best fit from an ordinary least squares regression between the medians of each posterior distribution is shown by the dotted line, with Pearson correlation coefficient, $R_p = 0.666$. Proline was not inferred using the Hearn data set but is annotated on the line of best fit for reference.

6.3.3 Sensitivity analysis

6.3.3.1 Antigen presentation is most sensitive to epitope parameters

A variance-based global sensitivity analysis, eFAST, was performed to identify the parameters most significantly affecting the presentation of a hypothetical epitope of length 8 residues (Figure 6.6). As expected, the presentation of the epitope was most sensitive to the off-rate of the corresponding peptide–MHC-I complex, particularly when measured through the primary sensitivity indices, S_i. The other parameters with notable sensitivity were the proteasomal supply rate, ERAP1 trimming rate, and the peptide–TAP binding rate of the epitope (significant at $p < 0.001$).

6.3.3.2 Aminopeptidases may play a destructive role in epitope generation

As well as the significant sensitivity to the epitope trimming by ERAP1, cytosolic aminopeptidase trimming of the 8mer was also predicted to have a significant ($p < 0.01$) effect on the 8mer presentation, albeit far less than the other epitope-associated parameters. This suggests that, for the parameter ranges considered, aminopeptidases may play a significant destructive role in over trimming epitopes.

6.3.3.3 Precursors significantly contribute to epitope presentation

A reduced effect on epitope presentation was predicted by our model for parameters corresponding to the N-terminally extended precursors (9mers to 16mers) through the primary sensitivity indices. The most notable of these parameters were the proteasomal production rate and the ERAP1 trimming rate of the 9mer. This suggests that the the trimming of this precursor in the ER can contribute to significant production of the epitope.

Further sensitivity to precursors was noted from the total sensitivity indices, S_T. In comparison to the primary indices, a much greater proportion of the sensitivity was seen to the proteasomal production and cytosolic aminopeptidase trimming rates of precursors, implying that N-terminus processing in the cytosol can also contribute to the generation of the epitope.

6.3.3.4 Longer precursors have little effect on presentation

The presentation of the epitope appeared to not be very sensitive to parameters associated with peptides longer than 11 amino acids. We found more generally that the presentation of 8mers to 12mers is predominantly only sensitive to the parameters associated with peptides between the epitope length and precursors of at most 3 amino acids from the N-terminus (Figure 6.7). This can be seen particularly clearly for the presentation of the 12mer, which showed very little sensitivity to any parameters associated with its 16mer precursor. Interestingly, the 12mer presentation was more sensitive to parameters associated with shorter peptides, suggesting that these peptides create competition for trimming by ERAP1, translocation by TAP, and loading to MHC-I.

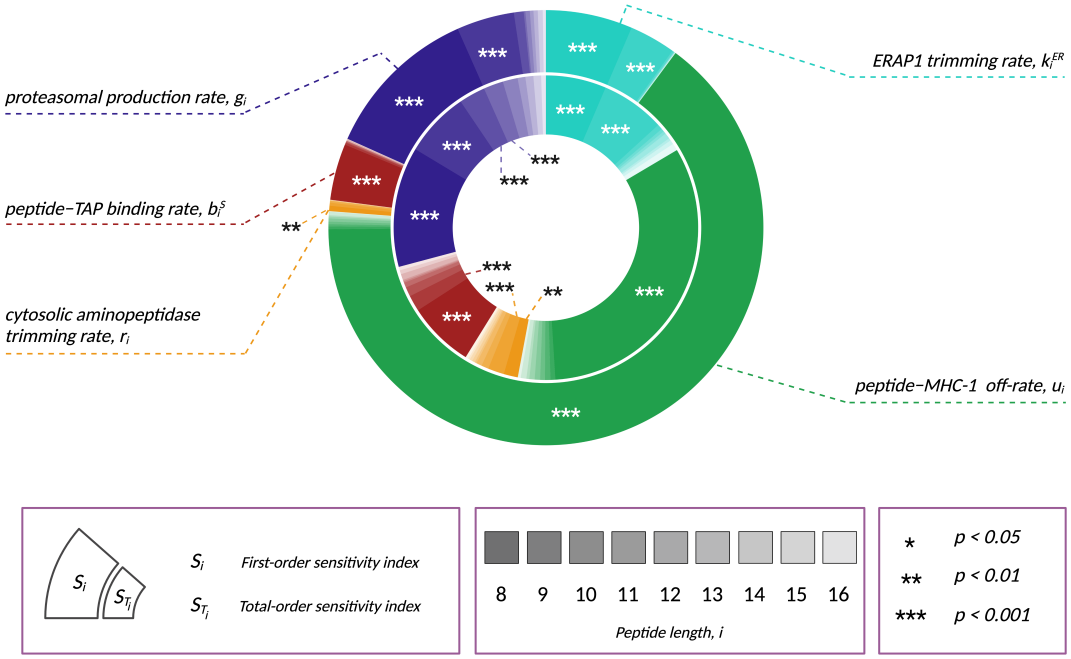


Figure 6.6. eFAST sensitivity indices compared across the 5 groups of parameter values. Shading intensity indicates the length of the peptide with which the parameter is associated. The outer doughnut graph shows the primary sensitivity indices, S_i, whilst the inner graph shows the total sensitivity indices, S_{T_i}. Sector width corresponds to the mean index value across the 5 repeats. Annotated p-values are the result of a one-tailed t-test between the parameter's indices and those of a dummy parameter.

6.4 Discussion

6.4.1 Possible reasons for discrepancies

6.4.1.1 Errors in parameter prediction

Although most of Hearn et al.'s observations were reproduced accurately by the mechanistic model, we found a small number of peptides for which a larger discrepancy existed between our predictions and the data. One explanation for this is that our predictive models for TAP and ERAP1 made inaccurate predictions for parameters corresponding to these peptides. However, the low sensitivity of the presentation of the 8mer to ERAP1 and TAP parameters for 9mers and 10mers seen in Figure 6.6, coupled with the accurate predictive model performances found through cross-

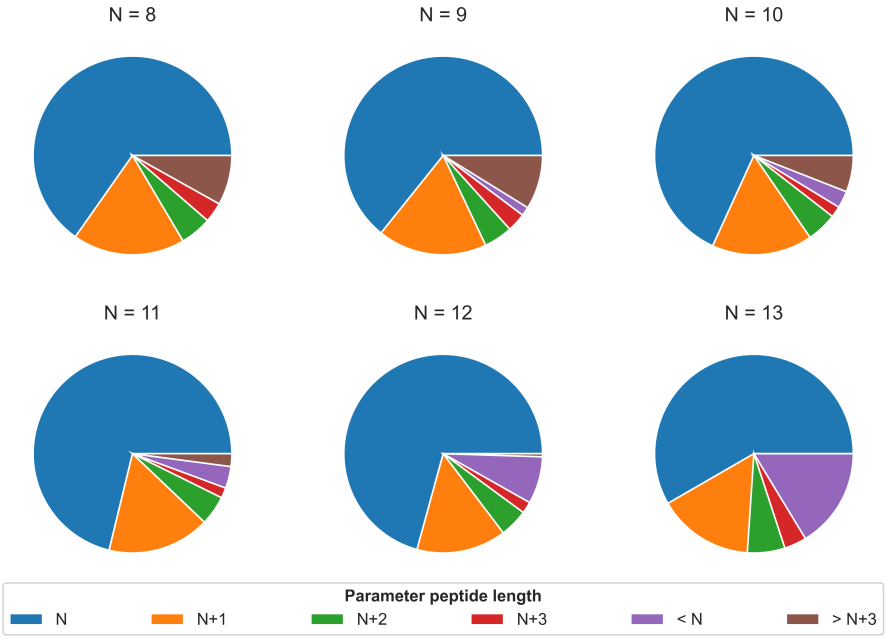


Figure 6.7. Total eFAST sensitivity indices for presentation of 8 to 13mers, grouped by parameter peptide length.

validation in Chapters 3 and 4, would imply that errors in these parameters alone would be unlikely to account for the discrepancies in our predictions.

6.4.1.2 Errors in parameter prediction

Alternatively, it is possible that the processing efficiency of these peptides is dependent on some other process that we have not included in our mechanistic model. One strong candidate for this is retrotranslocation from the ER to the cytosol — a pathway that is not currently included in our model. We chose to omit this pathway in order to create a more parsimonious model and because we did not anticipate that the retrotranslocation rate would be a practically identifiable parameter from the data available. Hearn et al. compared ER-targeted XXSL peptides in ICP47-treated HeLa cells (a competitive inhibitor for TAP, thus greatly reducing translocation efficiency) and in WT HeLa cells, concluding that certain peptides were likely to be retrotranslocated from the ER to the cytosol for trimming, before being re-translocated [71, 72]. Further research is required to understand the significance, if any, of this pathway in the generation of epitopes.

6.4.1.3 Ubiquitin removal in cytosol

Another possible explanation for discrepancies could be that ubiquitin is not cleaved uniformly efficiently from the N-termini of the different peptides. To test whether ubiqutin was removed equally efficiently across all N-termini, Hearn et al. repeated their study with methionine and alanine (MA) between the ubiqutin and the N-terminus extension. Although they found strong concordance between this data and the Ub XXSL data, valine was processed substantially more efficiently in the MAXXSL constructs than the XXSL, suggesting that VVSL supply in the cytosol may be limited due to inefficient ubiquitin removal. Hence, this could explain why our mechanistic model overestimated the presentation of SIINFEKL following the removal of valine from the N-terminus (VVSL).

6.4.2 Limitations of analysis

6.4.2.1 Peptide diversity

The similarity of the XXSL peptides is a potential limitation of the analysis in this chapter for the following reasons:

1. Many intermediate products (XSL) were in the training set for our ERAP1 predictive model.
2. The high sequence similarity might cause us to overlook substrate dependence of cytosolic aminopeptidases on the peptide sequence away from the N-terminus.
3. The peptides have similar MHC-I affinity/stability, so we are unable to validate predictions for better or worse binders.
4. The high sequence similarity means that we are only testing our TAP model predictions on a similar subset of potential substrates.

However, there are many reasons to believe that these drawbacks should not prove too restrictive to the validation and parametrisation of the mechanistic model. Firstly, although many of the XSL peptides were in the ERAP1 training set, no measurements for XXSL peptides were present in the literature. Hence, the mechanistic model's accuracy suggests that the ERAP1 model forms accurate predictions for these unseen peptides.

Furthermore, all peptides were unseen by the TAP model, which seems to have

largely formed accurate predictions of the binding affinities. This is supported by the observation that the predicted binding affinities correlate ($R_p = -0.674$) with the differences between the presentation of the ub XXSL peptides and the presentation of XXSL peptides targeted to the cytosol by fusing a signal sequence to the N-terminus [71] (Figure 6.8).

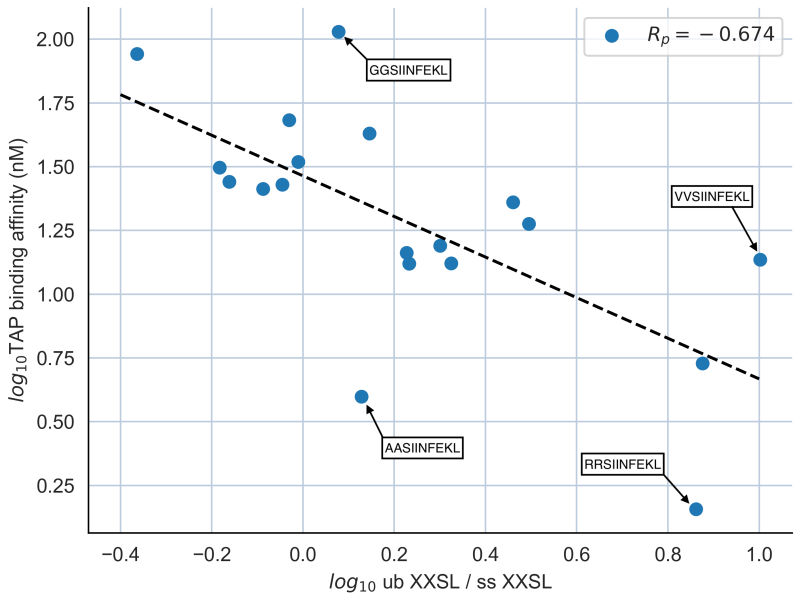


Figure 6.8. Predicted TAP binding affinities of XXSL plotted against fold difference in SIINFEKL presentation following cytosol- (ub) and ER-targeted (ss) XXSL processing. Outliers are labelled to show N-termini.

The similarity of the off-rates measured in our BFA decay assay restricts this parameter in our model validation to a narrow range of the possible values it could physically take. However, the peptide-loading stage of the model has already been validated on the *Dalchau* data set, with the role of tapasin well-characterised. Hence, this should not be a large concern. Furthermore, our sensitivity analysis shows that presentation of SIINFEKL should not be sensitive to the off-rates of its precursors.

Finally, the N-terminus dependence of trimming by cytosolic aminopeptidases inferred from XXSL trimming appears to correlate with Schatz et al.'s study of 9mer trimming by purified cytosol (Figure 6.5). This suggests that our parameters can be applied more generally to peptides of different sequences, despite having been inferred through the highly similar XXSL peptide set.

6.4.2.2 The role of ERAP2

Hearn et al. reported that the HeLa cell line used for the study did not express ERAP2 — the other aminopeptidase in the endoplasmic reticulum. The role of ERAP2 in the generation of epitopes is an area of ongoing research, so it is unclear how much its presence would be expected to influence the processing of the XXSL peptides. However, in a previous study, Hearn et al. had compared antigen processing in this HeLa cell line to a second cell line (H2-Kb transfected COS 7) that expressed both ERAP1 and ERAP2 [72]. They found a strong correlation between the processing of SIINFEKL precursors across the two cell lines ($R^2 = 0.7403$), with the only notable differences being a reduced efficiency removal of isoleucine and leucine in COS and a slightly more efficient removal of alanine. However, it should be noted that COS 7 is a cell line derived from African green monkeys (AGMs). The use of a different species will likely have introduced other confounding variables through inter-species heterogeneity in antigen processing machinery (e.g. cytosolic aminopeptidases or tapasin expression). Hence, it is difficult to interpret the extent to which the differences between the HeLa and COS 7 observations were caused by the presence of ERAP2 or another factor.

Nonetheless, the strong correlation suggests that the majority of N-terminus processing is carried out by ERAP1, and that there is consistency across multiple cell lines. High concordance in cytosolic aminopeptidase specificity across 7 different cell lines has also previously been reported, further strengthening our confidence that our mechanistic model should lead to accurate predictions across multiple cell lines [6].

6.4.2.3 Proteasomal predictions

The Hearn et al. study did not enable us to test the predictions of our proteasomal cleavage model. This is because we assumed a homogeneous supply rate of each peptide in the cytosol as we did not expect the short peptides (XXSL) to be processed significantly differently by the proteasome. Hence, the Hearn study gives us no way of validating the predictions of the probabilistic model developed in Chapter 2.

6.4.2.4 Lack of MHC-I diversity

In parametrising the extended *Dalchau* model, we used data from assays using an H2-Kb transfected .220 cell line [43]. Similarly, Hearn et al. studied an H2-Kb transfected HeLa cell line [71]. This meant that we could use the Dalchau parameters for modelling the Hearn study without any concerns about differences in tapasin

dependence or MHC-I supply. However, it also means that our mechanistic model has only been tested on a single HLA allele.

This might be a significant limitation in the modelling of highly tapasin dependent alleles. These alleles have an increased turnover rate of peptides in the endoplasmic reticulum. This might affect the availability of peptides for trimming by ERAP1, which could have a substantial effect on the predictions of precursor processing. It would therefore be prudent to validate the mechanistic model on a more tapasin dependent allele. To our knowledge no such data set currently exists in the literature.

6.4.2.5 Semi-quantitative data

Finally, Hearn et al. used fluorescence with a targeted monoclonal antibody to provide a readout for the level of SIINFEKL in H2-Kb on the HeLa cell surfaces. This shows us relative differences in abundance following the processing of different XXSL peptides but does not tell us the raw number of pMHC complexes on the cell surface. This uncertainty can be seen in the similarity of the posterior distributions of the scale factor (σ) and the peptide supply rate in the cytosol (g_P) in Figure 6.2: without knowledge of the exact number of complexes, we can explain the observed data using a higher peptide supply with a smaller scaling factor, and vice-versa.

This restricts our ability to assess whether the mechanistic model can accurately predict raw pMHC numbers or whether it can merely identify relative changes in presentation across different peptides.

6.4.3 Concluding remarks

In this chapter we have successfully inferred non-identifiable parameters from the extended *Dalchau* model presented in Chapter 5 and demonstrated the model's ability to accurately predict the processing of an epitope and different precursors by the class I antigen processing pathway. Moreover, we have inferred parameter values for epitope precursor trimming by cytosolic aminopeptidases. The resulting parameter values seem to be consistent with a relevant *in vitro* assay in the literature, giving us confidence that these parameters hold more generally [139].

Although our analysis is limited by the similarity of the XXSL peptides, many parts of our model have already been tested for their ability to form accurate predictions for a wide range of peptide sequences. Furthermore, despite the absence of ERAP2 in the HeLa cell line, Hearn et al.'s earlier comparison with an ERAP2-expressing cell

line (COS 7) suggests that the impact of omission of ERAP2 should not be very detrimental to the accuracy of our predictions of antigen processing and presentation.

From our sensitivity analysis, we concluded that the presentation of a peptide is predominantly sensitive to the production, processing and loading of itself, but also has some sensitivity to the production and processing of N-terminally extended precursors up to an extension length of approximately 3 amino acids. As we are interested in using the mechanistic model to study MHC class I epitopes, which are typically of lengths between 8 and 12 amino acids, we use a version of the presented mechanistic model including peptides between only 8 and 16 amino acids for the remainder of this thesis.

Because of the drawbacks mentioned in Section 6.4.2, we would ideally test our mechanistic model on a much larger and more diverse set of reported pMHC numbers. However, this type of data appears to be extremely limited in the literature, with only a handful of studies found reporting semi-quantitative (fluorescence) data for class I presentation, and even fewer reporting raw numbers of pMHC complexes. We instead consider in the next chapter whether the inclusion of our mechanistic model can enhance predictions of immunogenic CD8+ T-cell epitopes, under the assumption that efficacy on this task would be suggestive of accuracy its predictions of pMHC abundance. The use of such data should provide an indication of the mechanistic model's accuracy on a wider range of HLA types, peptide sequences, peptide lengths and cell lines.

Chapter 7

Prediction of immunogenic CD8+ epitopes through mechanism

7.1 Introduction

Upon developing and validating our mechanistic model of antigen processing in Chapter 6, the natural progression would be to evaluate the model's predictive accuracy using quantified peptide-MHC (pMHC) complex data from diverse cell lines and HLA allotypes. Regrettably, due to the technical complexity of the assays required, the literature is scant with datasets that provide quantitative measurements of class I epitopes. This scarcity significantly hampers our ability to robustly test the model's predictions.

Class I antigen presentation is a vital condition for the activation of naive CD8+ T-cells and the initiation of a cellular immune response. The degree of antigen presentation has been shown to have a positive correlation with the intensity of the subsequent T-cell response, as depicted in Figure 7.1 [168]. This observation aligns with the hypothesis that a certain threshold of T-cell receptor (TCR) stimulation is necessary to activate naive T-cells — an idea that is supported by *in silico* analysis of the causes of clinical failure of a cancer immunotherapy [24, 46].

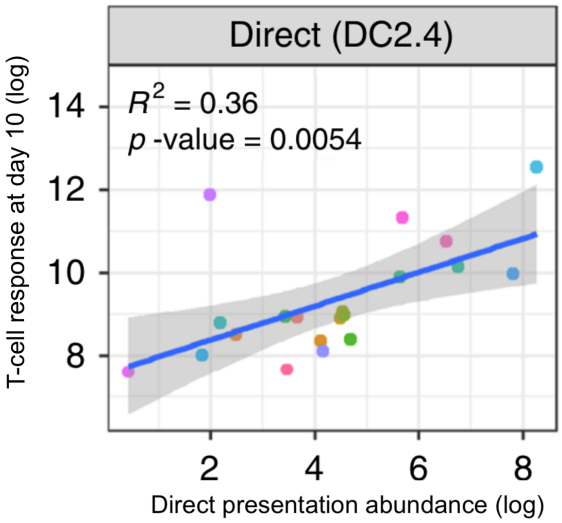


Figure 7.1. Correlation between direct presentation of influenza A epitopes and observed CD8+ T-cell response. Reproduced from Wu et al. [168].

In light of this, we were compelled to further assess our model's precision in forecasting the presentation of pMHC complexes by using its output to predict the activation of a CD8+ T-cell response. Machine learning predictions of class I presentation have already been used successfully to predict CD8+ T-cell immunogenicity in two prominent models in the literature: *PRIME-2.0* (previously *PRIME*) and *BigMHC* [10, 66, 140]. However, both of the presentation predictors in these models (*MixMHCpred-2.2* and *BigMHC EL* respectively) were trained using eluted ligand data from mass spectrometry. These immunopeptidomics datasets are binary labelled: a pMHC complex is either present amongst the eluted ligands (i.e. positive) or not (negative). Consequently, it should logically follow that antigen presentation predictors trained using these datasets are really learning to predict pMHC *presence*, rather than pMHC *abundance*. In the context of the relationship shown in Figure 7.1, this may lead to erroneous conclusions regarding immunogenicity, since pMHC may be presented in quantities of just one to tens-of-thousands of complexes [128].

We reasoned that, if our mechanistic model was indeed capable of forming accurate predictions of raw pMHC complex abundance, an immunogenicity predictor including the mechanistic output in its input features should be able to generate predictions of comparable or superior quality to the methods trained using only pMHC presence predictions. We call the resulting model and approach 'Prediction Of immunogenic Epitopes using Mechanistic modelling', or POEM for short.

In this chapter, we present the research behind the development of POEM. Our inspiration for the model's design and training largely came from *PRIME-2.0*, which is often considered to be state-of-the-art by the community. However, in addition

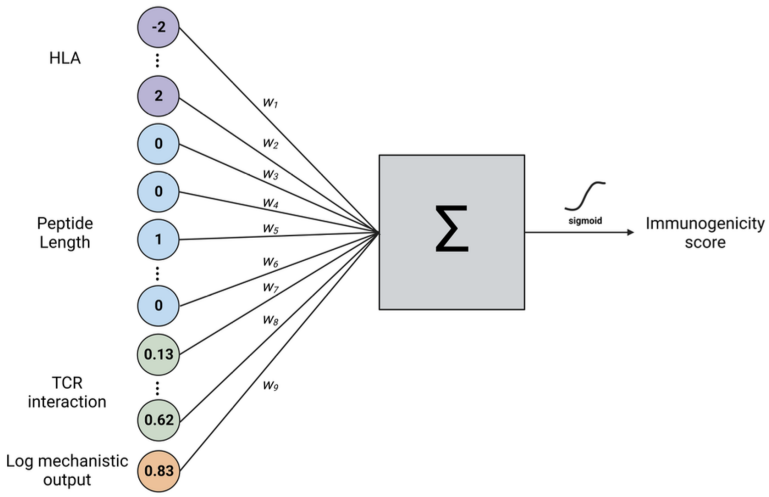


Figure 7.2. Schematic showing the 4 categories of POEM input feature. Logistic regressor feature weights are denoted by w_i. It should be noted that for the final version of POEM we apply principal component analysis to the input features, reducing the dimensionality (not shown).

to replacing the *MixMHCpred-2.2* output with our mechanistic prediction, we augmented and altered some of *PRIME-2.0*'s input features to try and further enhance performance. We conclude the chapter by benchmarking the resulting model against other prominent models from the literature and exploring the contribution of POEM's various input features to its efficacy.

7.2 Methods

7.2.1 POEM input features

The input layer to POEM consists of 4 categories of feature (shown in the schematic in Figure 7.2):

1. MHC-I allele representation
2. Peptide length encoding
3. Non-anchor peptide residue encoding
4. Mechanistic prediction of pMHC abundance

The choice of these features was inspired by the efficacy of *PRIME-2.0* and *NetTepi* in benchmarking studies from the literature [29]. The development of POEM and the nature of these input features are described further in the following subsections.

7.2.1.1 MHC-I pseudosequence

Although other immunogenicity predictors in the literature have usually omitted the restricting MHC-I allele from their input features, we surmised that the inclusion of a representation of the allele might enhance immunogenicity prediction due to contextualisation of predicted pMHC abundance and the interaction of key residues with the TCR.

We considered 3 different methods of representing the MHC-I allele sequence:

1. Using the pseudosequences proposed in *NetMHCpan* [114].
2. Using alternative pseudosequences proposed in *BigMHC* [10].
3. Using the full MHC-I sequence.

For each representation, we further compared the effect of encoding the sequence using the BLOSUM50 substitution matrix (as proposed in *NetMHCpan*) against using a sparse (one-hot) encoding of the sequence.

7.2.1.2 Peptide length encoding

To encode the peptide length, we used the same sparse encoding strategy as that used by Gfeller et al. in the development of PRIME-2.0. Specifically, the peptide length was encoded by a length 9 vector, representing a one-hot encoding of lengths 8 to 16 amino acids.

7.2.1.3 TCR contact sites

A lot of the peptide sequence information is already implicitly captured in the mechanistic model prediction through the sequence-specific processes of proteasomal cleavage, TAP translocation, cytosolic and ER aminopeptidase trimming, and MHC-I binding. In particular, the peptide residues at sites 1 to 3 from the N-terminus, along with the C-terminus, dominate most of these specificities. However, once presented, the peptide sequence must be recognised by a T-cell receptor (TCR), so a representation of the peptide sequence at these contact sites should be included in our input features to enable the prediction of the probability of this event occurring. We investigated three unique methods from the literature for this purpose.

1) PRIME-2.0 encoding

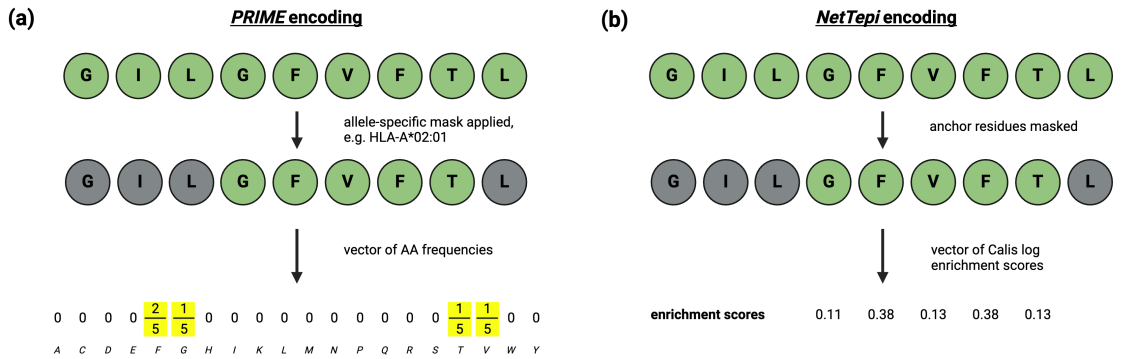


Figure 7.3. Schematic showing process of peptide encoding using (a) Gfeller et al.'s approach in PRIME and PRIME-2.0 [66, 140], and (b) an approach inspired by *NetTepi* [153].

In the development of *PRIME* (versions 1.0 and 2.0), Gfeller et al. used a set of HLA-I binding motifs to identify residues with a minimal impact on HLA-I affinity (MIA positions) by using a threshold information content of lower than 0.3 [66, 140]. This led to 6 groupings of alleles with unique MIA positions. For their immunogenicity predictor, Gfeller et al. then used a 20-dimensional vector containing the amino acid frequencies at these MIA positions as part of the input features. This process is summarised in the schematic in Figure 7.3.

2) NetTepi encoding

In the *NetTepi* algorithm, Trolle et al. included a T-cell propensity score in their T-cell epitope prediction [153], adapting earlier work by Calis et al. [32]. Calis et al. used a set of dataset of 9mers with validated immunogenicity status to calculate the enrichment of each amino acid at sites P1 to P9. The authors then used these enrichment scores to calculate an immunogenicity score for any 9mer by summing the log enrichment scores at all non-anchor sites in the peptide, weighted by the importance of each position (established by looking at the Kullback-Lieber divergence of the immunogenic and non-immunogenic distributions).

Trolle et al. then extended this analysis to accommodate peptides of lengths 8 to 14 amino acids by padding 8mers with an unknown amino acid (X) and using the approximation method of Lundegaard et al. to represent an N-mer with 6 different 9mers by removing sequences of length $N - 9$ between sites P4 and PN-1 [104].

We adapted Trolle et al.'s propensity score to construct a length 5 representation of the TCR contact sites containing log enrichment scores of the amino acids at P4 to P8 for a 9mer. We decided not to include the positional weightings used in *NetTepi* under the assumption that these weightings should be learned in the downstream classification task. For 8mers, we included a padding token at P4, whilst for longer

peptides, we took the arithmetic mean of the representations of 6 different 9mers generated using Lundegaard et al.'s method. This is summarised for an example of a 9mer in Figure 7.3.

3) ESM-2 embedding

Finally, we considered an embedding of the peptide sequence using representations from pre-trained protein language models. For this purpose, we used Meta's Evolutionary Scale Modelling (*ESM-2*) pre-trained model with 8 million parameters [101]. Each peptide was converted to a 320-dimensional numerical representation by taking the mean of the final transformer layer (as recommended by Meta). We also investigated the effect of masking the non-MIA positions (as defined by Gfeller et al.) using masking tokens from the *ESM-2* vocabulary before embedding.

For each of the peptide encoding strategies, we re-trained POEM and tuned the number of principal components from scratch before statistically comparing the models' 10-fold cross validation performances through a paired Wilcoxon signed rank test.

7.2.1.4 Mechanistic prediction of antigen presentation

The final feature in POEM's input is the product of the research presented in previous chapters in this thesis. We extended the system of ordinary differential equations used to simulate the Hearn et al. study in Chapter 6 to model the movement of 16mers to 8mers through the antigen processing pathway, where the studied epitope is one of these peptides.

The supply of each of these peptides in the cytosol was replaced by the output of the probabilistic model of proteasomal cleavage presented in Chapter 2. We assumed an approximate supply of 1 copy of the source protein per second to the proteasome and took the product of this with the predicted proteasomal production probabilities to determine the cytosolic supply of each peptide per second. This estimate was based upon estimated transcription and translation rates in HeLa cells from Hausser et al., suggesting a typical protein synthesis rate on the order of 10^3 proteins per hour [70]. We assumed that the synthesis rate would likely be higher in cancerous or pathogen-infected cells, so settled upon an estimated rate of 1 protein per second.

Cytosolic aminopeptidase activity was estimated for each peptide using the trimming rates determined in Chapter 6. TAP translocation of each peptide was predicted using *PanTAP* (Chapter 3) and subsequent N-terminus processing by ERAP1 was estimated using the predictive model in Chapter 4.

Remaining parameters were all fixed at the values in Table A.1 (in Section A.3.2), with the exception of the peptide-MHC binding rate, b. This parameter was adjusted from its H2-Kb value for each MHC-I allele using the tapasin dependence scores reported by Bashirova et al. and the calibration curve derived in Figure 5.7. Where tapasin dependence scores were not reported by Bashirova et al., the arithmetic mean of reported tapasin dependence of alleles of the same supertype was taken as an estimate of the unknown tapasin dependence.

Peptide–MHC off-rates were predicted using *NetMHC-4.0* (or *NetMHCpan-4.1* on alleles for which *NetMHC-4.0* could not make predictions). These models return a predicted binding affinity in units of nanomolarity. We converted these binding affinities into units of cellular concentration (molecules per cell) using the method in Section 5.2.2.2. The resulting binding affinity (K_D) was then related to the off-rate and binding rate via the formula for the dissociation constant: $K_D = u_i/b$, enabling the calculation of the off-rate (u_i).

Other immunogenicity prediction models (e.g. *PRIME*, and *HLAthena*), scaled their predictions of antigen presentation by ranking presentation scores for each allele against random natural peptides sampled from the human proteome. This was done to facilitate comparison between different alleles. We decided against such an approach, reasoning that higher levels of antigen presentation should enable the antigen presenting cell to bind to more TCRs, thus increasing the chance of CD8+ T-cell activation, regardless of the allelic distribution of pMHC abundance. Instead, we used the log-scaled quantity of presented pMHC complexes predicted by simulating the mechanistic model to equilibrium.

7.2.2 POEM training

7.2.2.1 PRIME-2.0 training set

To train POEM, we used the dataset of 6,680 neoantigens compiled by Gfeller et al. to train *PRIME-2.0* [66]. This dataset was formed by consolidating data from several neo-antigen studies with neo-epitope data from IEDB, removing overlap. This resulted in 596 experimentally verified immunogenic neo-epitopes and 6,084 verified non-immunogenic peptides, covering lengths of 8 to 14 amino acids, and 66 different HLA alleles.

Gfeller et al. included randomly sampled natural peptides from the human proteome to act as negative decoys in the model training. We decided against this approach, reasoning that the 6,680 non-random peptides had been identified in their various

studies due to their predicted MHC-I binding affinity, so should be distinguishable from the decoy peptides based on binding affinity alone. We did not want to train POEM to predict binding affinity but instead to predict pMHC complex recognition by naïve T-cells.

7.2.2.2 Data pre-processing

The *PRIME-2.0* training dataset contains a list of mutant peptides paired with their restricting HLA allotype, their source protein, and their immunogenicity status. We downloaded FASTA files containing sequence data for the source proteins from the reference human proteome on UniProt (UP000005640) and replaced the wild-type peptides with the mutants [148]. These mutated FASTA files were then passed to the proteasomal cleavage prediction algorithms, forming the start point for the mechanistic modelling pipeline. From these FASTA files, we were also able to extract N-terminally extended precursor peptide sequences for each peptide up to length 16 amino acids for prediction of parameter values in the mechanistic model.

7.2.2.3 Model training

We employed two widely recognised models to train our classifier: logistic regression and a multi-layer perceptron (MLP). Logistic regression serves as a linear classifier and is frequently utilised as a benchmark in machine learning due to its simplicity and effectiveness in binary classification tasks. On the other hand, the MLP, a basic form of neural network that incorporates a single hidden layer, extends our capability to identify complex, non-linear relationships among the input features. This added complexity has the potential to enhance model performance by capturing intricate patterns in the data. However, it also introduces the possibility of overfitting to the training dataset, a scenario where the model learns the noise in the data rather than the underlying pattern, affecting its ability to generalise to unseen data.

We trained each of the classification models on the PRIME-2.0 training set, applying input feature scaling through the use of scikit-learn's StandardScaler [125]. Additionally, we incorporated principal component analysis (PCA) into the logistic regression model's preprocessing steps, a decision informed by its positive impact on performance as evidenced in a cross-validation study. The optimal number of principal components was dynamically determined for each POEM model iteration, based on the configuration that achieved maximal accuracy in a 10-fold cross-validation. In contrast, the application of PCA did not yield a performance advantage for the MLP model, leading to its exclusion from the MLP's training process.

We trained the MLP with a single hidden layer containing 32 nodes, which was found to be the best-performing architecture of an initial set of candidate models with 8, 16, 32, 64, 128 and 256 hidden nodes. To avoid over-fitting, we randomly held out 15% of the training set to use as a validation set for early-stopping of the MLP training. All MLP training was carried out using the *Tensorflow* library in Python [1], whereas logistic regressor training was implemented through the scikit-learn package [125].

7.2.3 POEM testing

Having trained POEM on the entirety of the the PRIME-2.0 training set, we then sought to compare its performance to other prominent algorithms in the literature across a range of different challenges.

7.2.3.1 GBM dataset

To test POEM's ability to form accurate predictions on a single study and cancer, we used a previously presented dataset of 124 glioblastoma multiforme (GBM) peptides with experimentally validated immunogenicity [98]. This dataset was constructed using sequencing and expression data from the tumours of four HLA-A2 glioblastoma patients. A version of the MuPeXI pipeline was used to predict 153 HLA-A2-restricted neoantigens, of which 29 were predicted to be poor binders to HLA-A*02:01 so were dropped. Of the remaining peptides, 25 were characterised as immunogenic and 99 as non-immunogenic by functional T-cell assays [106].

In the same way as with the *PRIME-2.0* dataset, we downloaded FASTA files for the source genes provided in the original study from UniProt and replaced the wild-type peptides with the mutants [148]. These modified FASTA files served as the input for proteasomal cleavage prediction algorithms, marking the initial step in our mechanistic modeling approach.

The restriction to a single study of origin and a single HLA allele (A0201) mimics a plausible clinical situation in which POEM might feasibly be used to rank a shortlist of identified mutations in terms of predicted immunogenicity (e.g. for a personalised cancer immunotherapy). The use of a single cancer also alleviates the issue of differences in protein expression known to exist in different cancers [67], although it should be noted that there can be considerable heterogeneity in antigen expression across a single tumour.

7.2.3.2 BigMHC IEDB dataset

To assess POEM's capability in precisely forecasting the immunogenicity of pathogen-derived epitopes, we employed a dataset previously compiled by Albert et al. for evaluating the *BigMHC* tool [10], henceforth referred to as the 'IEDB pathogen dataset'. This comprehensive dataset spans an array of 96 pathogens and includes 86 distinct HLA allotypes, featuring 2,345 peptides of which 1,701 are classified as immunogenic and 644 as non-immunogenic.

To simulate POEM's mechanistic module, we downloaded reference protein sequences for the various pathogens from UniProt and located the peptides within these FASTA files [148]. The corresponding FASTA files were then used for proteasomal cleavage prediction. N-terminally extended precursors up to 16mers were also extracted from the source protein sequence for use in the mechanistic model.

7.2.3.3 SARS-CoV-2 dataset

To enhance our exploration of POEM's capacity to pinpoint immunogenic epitopes of pathogens and to mitigate the impact of confounding factors, we concentrated our efforts on a singular disease. The choice of SARS-CoV-2 as our focus was driven by the extensive availability of data shared during the COVID-19 pandemic.

We extracted all MHC-I restricted epitopes documented with T-cell response data (either positive or negative) in the Immune Epitope Database (IEDB) up to 15 August 2023. We categorised each epitope as immunogenic if it had at least one positive assay result, with the rest deemed non-immunogenic. This process yielded a dataset comprising 2,526 unique peptide-HLA combinations, including 1,908 epitopes identified as immunogenic and 618 as non-immunogenic, derived from a total of 5,964 assay reports. The dataset encompasses 42 distinct HLA allotypes and spans 13 different proteins within the SARS-CoV-2 proteome. However, we chose to focus on 6 specific alleles, comprising 67.9% of the dataset, in order to draw more statistically robust conclusions of efficacy and test POEM's ability to accurately rank epitopes within single HLA allotypes.

Across these 6 alleles, we found considerable heterogeneity in the proportion of immunogenic peptides. Only 34.8% of the peptides associated with HLA-A*03:01 were labelled as immunogenic, whereas 85.4% of HLA-A*01:01 restricted peptides had at least one positive assay result. We also found moderate heterogeneity in the source proteins of the peptides in these 6 groups. For instance, 70.4% of peptides in the HLA-A*03:01 group were from the replicase polyprotein 1ab ORF, whereas only 45.9% of peptides in the HLA-A*24:02 group were from this ORF. However,

betweeen 70% and 80% of peptides in each group were from replicase polyprotein 1ab or the spike protein.

7.2.4 POEM benchmarking

To contextualise the performance of our model on these training and test sets, we compared POEM's predictions with those of other prominent models in the literature. For direct comparison, we chose the subset of models used by the authors of *BigMHC* to benchmark their model.

These methods were trained using combinations of measured *in vitro* MHC-I binding affinity (BA), mass spectrometry identified eluted ligands (EL), and reported CD8+ immunogenicity (IM). All methods except for *PRIME2.0* predict the probability of a peptide being a naturally presented MHC-I ligand (EL score). *BigMHC* uses its EL score (*BigMHC EL*) for a separate predictor of immunogenicity (*BigMHC IM*), whereas *PRIME-2.0* uses the EL score of *MixMHCpred-2.2* to form its immunogenicity predictions. The methods and their main features are summarised in Table 7.1.

Table 7.1. Predictive Models of CD8+ Epitope Immunogenicity

Model	Description
MixMHCpred-2.2 [15]	Uses positional weighted matrices (PWMs) trained on EL data to predict MHC-I ligands.
NetMHCpan-4.1 [130]	An artificial neural network trained on mass spectrometry eluted ligands and *in vitro* binding affinity assays. Returns a presentation score corresponding to the probability of a peptide being presented by any given MHC-I allele.

Table 7.1 – continued from previous page

Model	Description
MHCflurry-2.0 [118]	A novel predictor of binding affinity (neural network, trained with BA and EL data) and antigen processing efficiency (neural network, trained on EL data and output of binding affinity predictor). These two scores are then combined via logistic regression to predict EL probability.
MHCnuggets-2.4.0 [143]	A long short-term memory (LSTM) neural network trained MHC class I and class II peptide bindings, facilitating comprehensive epitope mapping for CD8+ T cell responses.
TransPHLA [39]	A transformer-based model that predicts peptide-HLA binding and uses attention scores to optimally design mutated peptides. Trained using BA and EL data. Final softmax layer removed for this study, as recommended by Albert et al. [10], to enable AUROC and AUPR calculation.
HLAthena [136]	Uses 3 single-layer neural networks trained on MS data to predict presentation.
PRIME-2.0 [66]	A multi-layer perceptron (MLP) using *MixMHCpred-2.2* rank, peptide length, and non-anchor peptide residues to predict CD8+ immunogenicity. Trained on IM data to infer the mechanisms of TCR recognition of pMHC complexes.

Table 7.1 – continued from previous page

Model	Description
BigMHC [10]	Uses EL data to train presentation predictor (*BigMHC EL*) before transfer learning on IM data to predict immunogenic CD8+ T cell epitopes (*BigMHC IM*).

7.2.5 Performance evaluation

To assess the predictive efficacy of the POEM model alongside other established models within the literature, our evaluation employed two widely recognised metrics: the area under the receiver operating characteristic curve (AUROC) and the area under the precision-recall curve (AUPR).

The AUROC metric quantifies the likelihood that a model correctly ranks a randomly selected positive instance higher than a negative one, serving as a commonly understood benchmark in the field. Despite its prevalence, AUROC's reliability as a performance indicator may diminish in situations characterised by significant class imbalance, potentially misleading the assessment of model effectiveness. Under such circumstances, the AUPR metric emerges as a preferable alternative, owing to its focus on the precise identification of positive instances within the minority class.

Given the pronounced class imbalance in the neoantigen datasets (PRIME-2.0 and GBM), AUPR emerges as the more fitting metric, while the pathogenic datasets, predominantly comprising positive instances, align more closely with the application of AUROC for performance evaluation. The ensuing analysis in our results section explores the performance of POEM and comparable classifiers through both metrics, offering a comprehensive overview of their predictive capabilities.

Furthermore, where cross-validation has been implemented, we assess the comparative performance of the classifiers via the paired Wilcoxon signed-rank test. This non-parametric statistical method calculates a p-value, reflecting the likelihood that the observed differences between the two groups of results are attributable to random variance, thereby providing a robust statistical basis for performance comparison.

7.3 Results

7.3.1 POEM model development

7.3.1.1 Comparison of proteasomal prediction algorithms

We simulated the mechanistic model using all 6 of the proteasomal prediction algorithms considered in Chapter 2 and evaluated performance on the PRIME-2.0 training set, assuming that an increase in prediction accuracy of antigen presentation would result in an improvement in classification performance.

Across the 6 algorithms, we found only modest differences in AUROC and AUPR, shown in Table 7.2. The epitope version of the *Pepsickle* algorithm resulted in predicted pMHC abundances with the highest AUROC and AUPR with the PRIME-2.0 dataset.

We then trained a logistic regressor using the mechanistic model outputs with each of the proteasome algorithms, along with a BLOSUM50-encoded *NetMHCpan* pseudosequence, a sparse length encoding, and a *NetTepi* peptide encoding. We did not find a significant difference between the performance of any pair of these classifiers on a 10-fold cross-validation, but again found that the epitope version of the *Pepsickle* algorithm resulted in the highest mean AUROC and AUPR (Figure 7.4)..

We concluded that the choice of proteasomal cleavage prediction algorithm has only a minor effect on the mechanistic model output and downstream classification task. In absence of any evidence to suggest otherwise, and because of its efficacy on the prediction of ovalbumin digestion products in Chapter 2, we decided to use the epitope version of the *Pepsickle* algorithm for the remainder of our analysis in this chapter.

Algorithm	AUROC	AUPR
Pepsickle (epitope)	0.601	0.115
Pepsickle (in vitro)	0.600	0.114
Pepsickle (in vitro 2)	0.598	0.114
NetChop	0.601	0.114
NetChop 20s	0.599	0.113
BiLSTM	0.598	0.114

Table 7.2. Comparison of mechanistic model performance on on PRIME-2.0 training set.

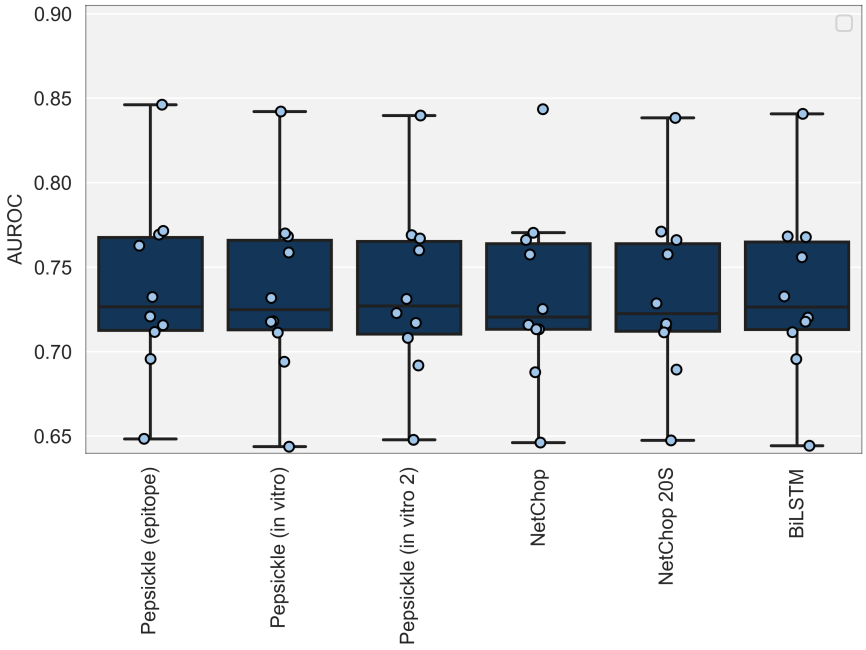


Figure 7.4. Classification performance of 6 POEM models trained using different proteasomal cleavage algorithms compared by 10-fold cross-validation on *PRIME-2.0* dataset. No significant difference was found between any pair of models by a two-tailed paired Wilcoxon signed rank test (at $p = 0.10$ threshold).

7.3.1.2 Comparison of MHC-I sequence representations

For the input features to our predictor, we compared 3 different representations of the restricting MHC-I allele and 2 methods of encoding the amino acid sequence with a numerical vector. For each representation and encoding, we trained a logistic regressor using the *PRIME-2.0* training set and evaluated performance via a 10-fold cross-validation. In addition to the MHC-I representation, we included a sparse length encoding, mechanistic model prediction, and *NetTepi* peptide encoding in the input features.

We did not find a significant difference between the AUROC of the BLOSUM50 encoded *NetMHCpan* pseudosequence and any of the other 5 methods (Figure A.1). We did, however, find that the AUPR of this method was significantly higher than the sparse encoding version of the *NetMHCpan* pseudosequence, or either encoding of the *BigMHC* pseudosequence (Figure 7.5). The BLOSUM50 encoded full MHC-I sequence led to a slightly higher AUPR on the cross-validation performance. However, this was neither a a significant increase on the *NetMHCpan* pseudosequence, nor did we find any significant difference in predictions on the test datasets. Hence, we decided to use the BLOSUM50 encoded *NetMHCpan* pseudosequence in order to keep our input feature dimensionality low, thus improving model training and prediction speed.

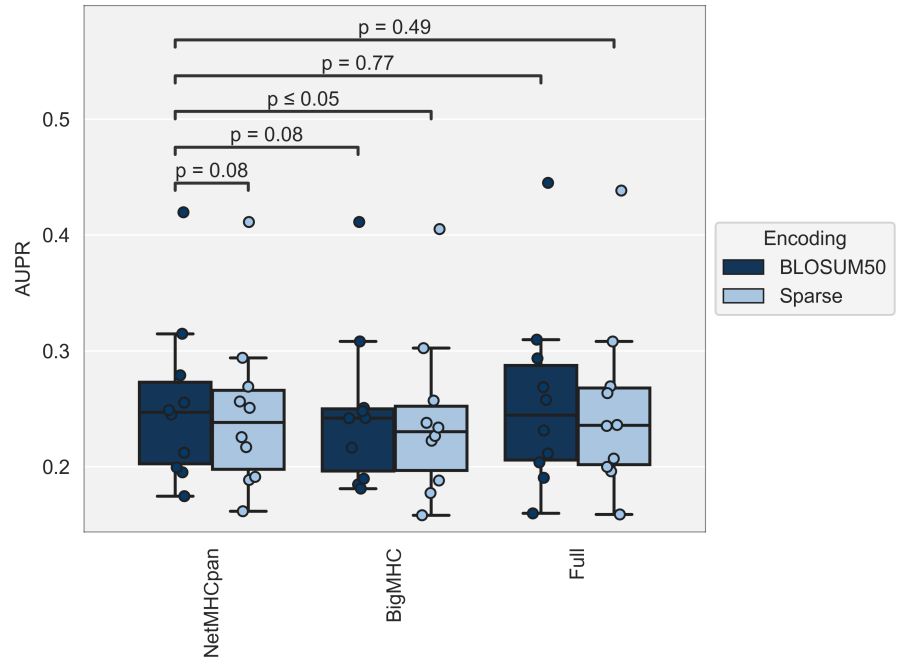


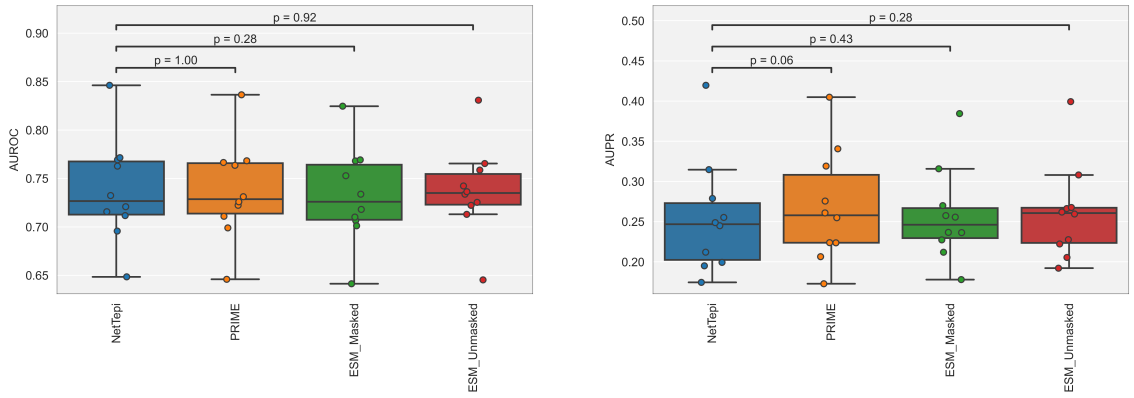
Figure 7.5. MHC-I representations and associated amino acid encoding strategies compared by 10-fold cross-validation on *PRIME-2.0* dataset. Annotated p-values indicate the result of a two-tailed paired Wilcoxon signed-rank test between AUPR scores.

7.3.1.3 Comparison of peptide sequence representations

We compared 4 different approaches to include epitope sequence information in our model input features: encodings inspired by *NetTepi* and *PRIME-2.0*, and embeddings using *ESM-2*, with or without masking of anchor residues.

For each method, we trained a logistic regressor using the *PRIME-2.0* training set and evaluated performance by means of a 10-fold cross-validation (shown in Figure 7.6). We did not find a significant difference in AUROC or AUPR between *NetTepi* and any of the other 3 representations (at $p = 0.05$ level), although the *ESM-2* embedding with no masking yielded the highest median AUROC and AUPR, and the *PRIME-2.0* was close to having a significantly higher AUPR than the *NetTepi* encoding ($p = 0.06$).

However, when running the GBM, IEDB and COVID test sets through the resulting models, we found notable differences in performance across the 4 peptide representations, particularly on the COVID set (shown in Figure A.2). Whilst no one method performed best across the 6 alleles of focus, we concluded that the model resulting from the *NetTepi* encoding was the most consistently accurate across each allele. Hence, we decided to use this peptide encoding in our final model.



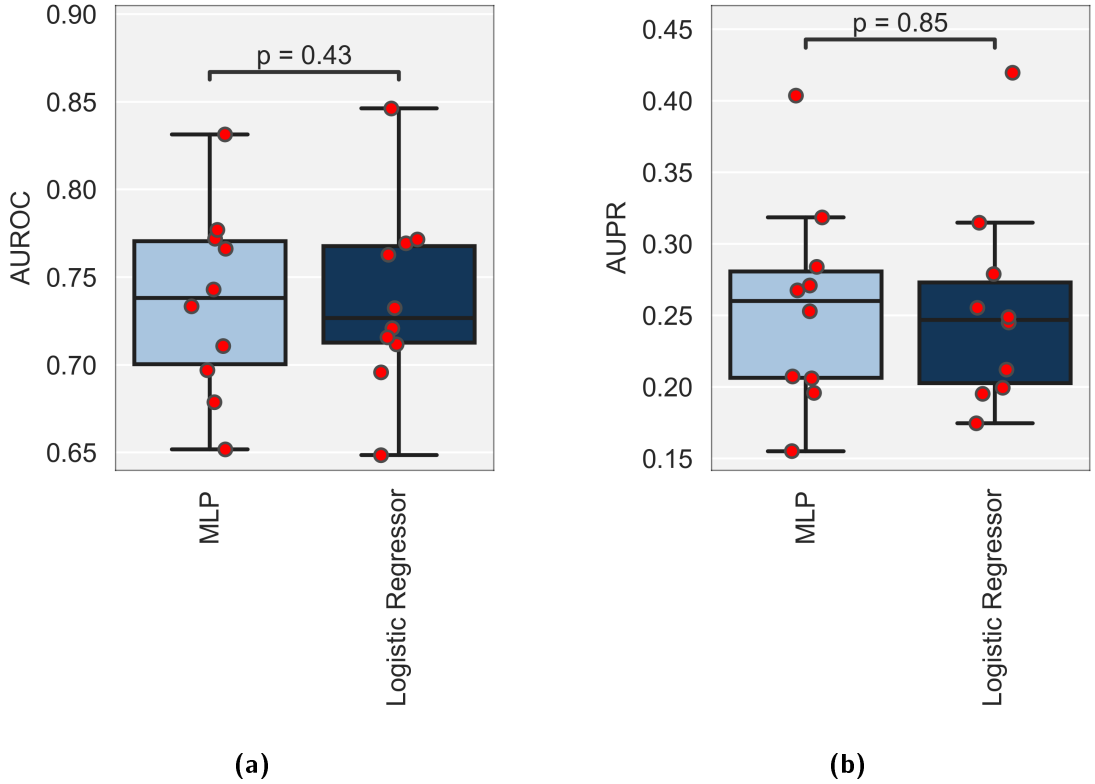
(a) **(b)**

Figure 7.6. Peptide encoding methods compared by 10-fold cross-validation on *PRIME-2.0* dataset. p-values indicate the result of a two-tailed paired Wilcoxon signed-rank test.

7.3.1.4 Logistic regression vs. multi-layer perceptron

With high accuracy already established using a logistic regressor, we investigated whether a further enhancement in efficacy could be obtained through the use of an MLP, enabling the incorporation of non-linearities in the input features.

Although we observed a difference in performance between the logistic regressor and the MLP, with the MLP seemingly having a higher median AUROC and AUPR across the 10 folds, these differences were not found to be significant using a paired Wilcoxon signed rank test ($p = 0.43$ and $p = 0.85$ respectively, two-tailed test). Due to the lower computational overheads and greater interpretability of the logistic regressor, we therefore resolved to use the logistic regressor version of POEM for the remainder of our analysis. However, the performance of the trained MLP version of the model is shown in the analysis of the test sets for comparison.



(a) **(b)**

Figure 7.7. Multi-layer perceptron performance compared to logistic regressor performance through 10-fold cross-validation on *PRIME-2.0* training dataset. Predictive performance is given in terms of (a) AUROC, and (b) AUPR. A paired Wilcoxon signed-rank test was used to test for significance in the differences between the models and the resulting p-values are indicated on each plot.

7.3.2 POEM performance

Having established appropriate choices of input feature, we trained POEM using the *NetMHCpan* pseudosequence, *NetTepi*-inspired peptide sequence representation, sparse length encoding, and mechanistic pMHC abundance prediction.

7.3.2.1 POEM improves prediction accuracy on PRIME-2.0 training set

We used a stratified 10-fold cross-validation to compare the performance of POEM to the other models in Table 7.1. To obtain an unbiased measure of model performance, we re-trained *PRIME-2.0* and *BigMHC* for each fold, removing any overlap between the respective models' training sets and the current test set. *BigMHC* could be re-trained using its source code. For *PRIME-2.0*, we re-trained the model using a logistic regressor rather than an MLP to avoid tuning hyperparameters and potentially using a different approach to the authors. This was shown by Gfeller et al. to have

no significant impact on *PRIME-2.0*'s efficacy when predicting without the presence of random decoys [66].

We found that POEM had a significantly higher AUROC and AUPR (shown in Figure A.3) than all other models considered ($p \leq 0.01$ threshold). The next best performing methods were the two models trained to specifically predict CD8+ immunogenicity: *BigMHC IM* and *PRIME-2.0*. However, many of the methods trained to predict eluted ligands also scored highly, particularly the *MHCflurry-2.0* algorithm. This demonstrates a strong correlation between antigen presentation and immunogenicity.

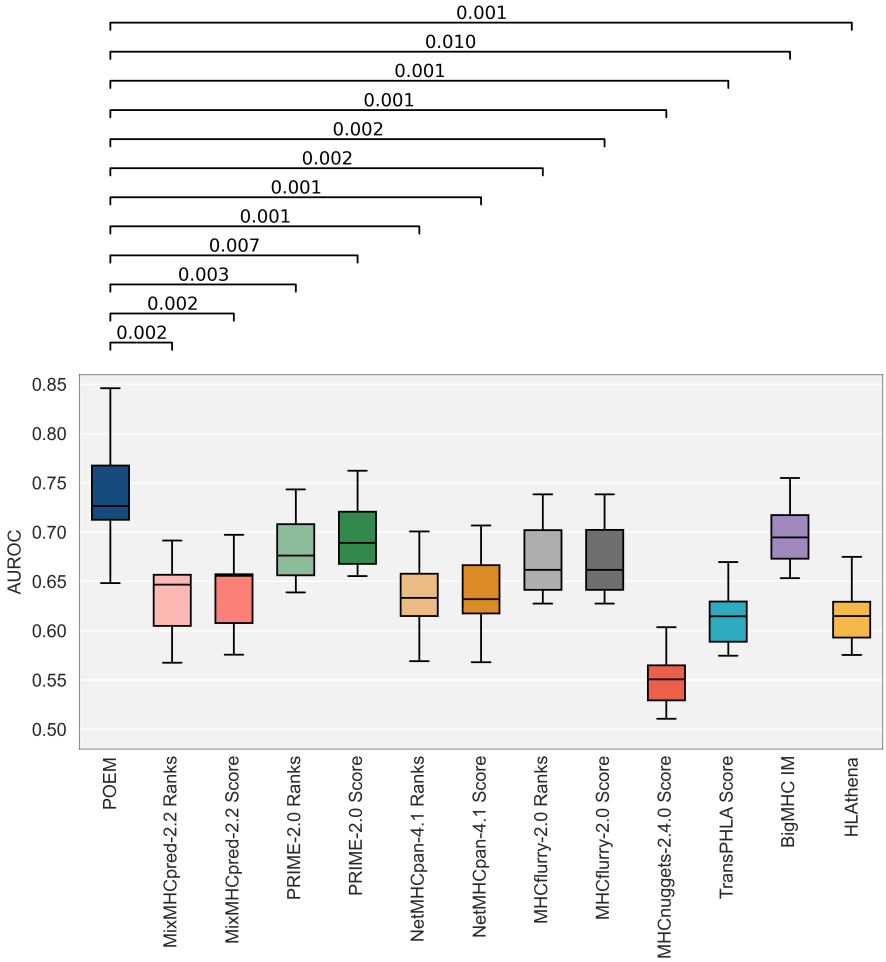


Figure 7.8. Performance of POEM compared with the models in Table 7.1 via a 10-fold cross-validation and using the AUROC metric. A one-tailed paired Wilcoxon signed-rank test was used to test whether the AUROC of the other models was significantly lower than that of POEM. The resulting p-values are shown above the plot.

7.3.2.2 POEM accurately identifies immunogenic GBM neoepitopes

Having found that POEM classifies immunogenic neoantigens across a combination of cancers, studies and HLA allotypes with greater accuracy than comparable methods, we tested its accuracy on a set of experimentally validated neoantigens from a single cancer type (GBM), study, and HLA allotype (HLA-A*02:01). The predictive efficacy of POEM was marked by a higher AUROC of 0.776 and AUPR of 0.501 than the next best-performing methods, *BigMHC* (AUROC = 0.749) and *TransPHLA* (AUPR = 0.472) respectively (shown in Figure 7.9). Notably, the MLP variant of POEM demonstrated enhanced accuracy over the logistic regression version, attaining an AUROC of 0.801 and an AUPR of 0.579.

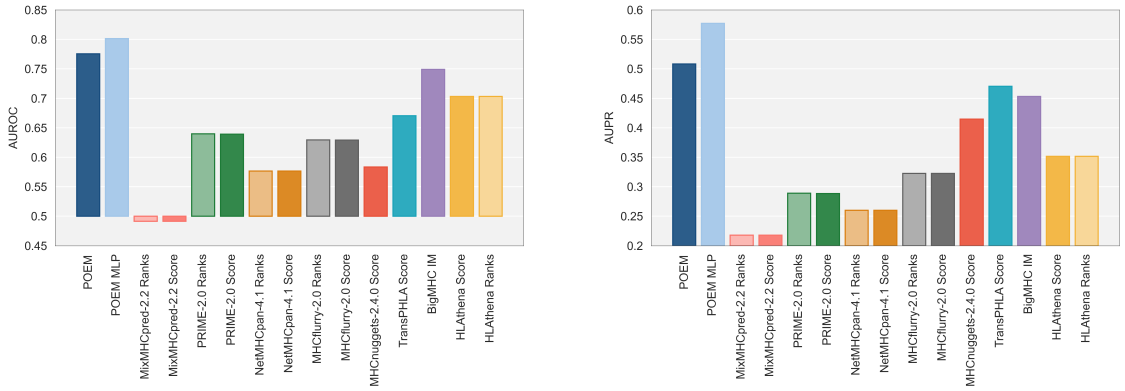


Figure 7.9. POEM predictive performance benchmarked against other prominent immunogenicity predictors for a set of 124 HLA-A*02:01 restricted epitopes derived from glioblastoma multiforme (GBM) and reported in the literature [97].

7.3.2.3 POEM predicts pathogenic epitopes

We then tested the model's ability to accurately identify immunogenic pathogen-derived epitopes by using *BigMHC*'s IEDB dataset. Again, POEM was the best-performing of the methods considered, with an AUROC of 0.662 and an AUPR of 0.821 far exceeding the accuracy of *PRIME-2.0* (AUROC = 0.595, AUPR = 0.789) and *BigMHC* (AUROC = 0.587, AUPR = 0.787). The MLP variant of POEM demonstrated similar performance to the logistic regression version, attaining an AUROC of 0.668 and an AUPR of 0.821.

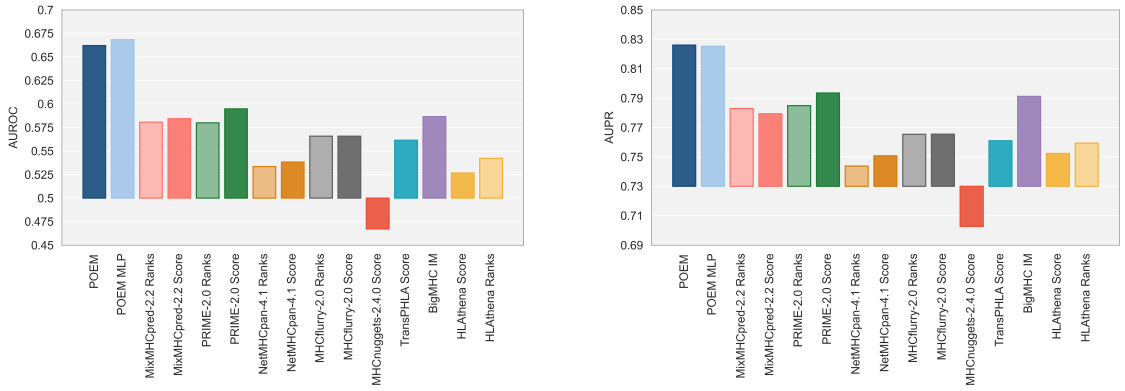


Figure 7.10. POEM predictive performance benchmarked against other prominent immunogenicity predictors for the set of pathogenic peptides consolidated from IEDB and reported in *BigMHC* [10].

7.3.2.4 POEM accurately identifies SARS-CoV-2 epitopes across a range of HLA types

In order to focus on a single infectious disease, we constructed our own dataset of SARS-CoV-2 epitopes from IEDB and benchmarked POEM's predictions. We chose to concentrate our analysis on the 6 HLA alleles with the greatest number of associated peptides in the dataset: A0101, A0202, A0301, A1101, A2402, and B0702. The AUROC for each method across these alleles is shown in Figure 7.11 and the AUPR in Figure A.4.

We found that POEM had the highest AUROC and AUPR of any method on A0101, A0201, A1101, A2402 and B0702, whilst on A0301, *PRIME-2.0* was the best-scoring method and was the most consistently accurate of the other methods across the 6 alleles. As the other model trained to specifically predict immunogenicity, *BigMHC* performed inconsistently across the alleles and was worse than random chance (AUROC = 0.5) for A2402 and B0702.

All models performed better on A0201 than any other allele. The MLP version of POEM displayed similar accuracy for 4 of the 6 alleles but was substantially worse on A2402 and B0702.

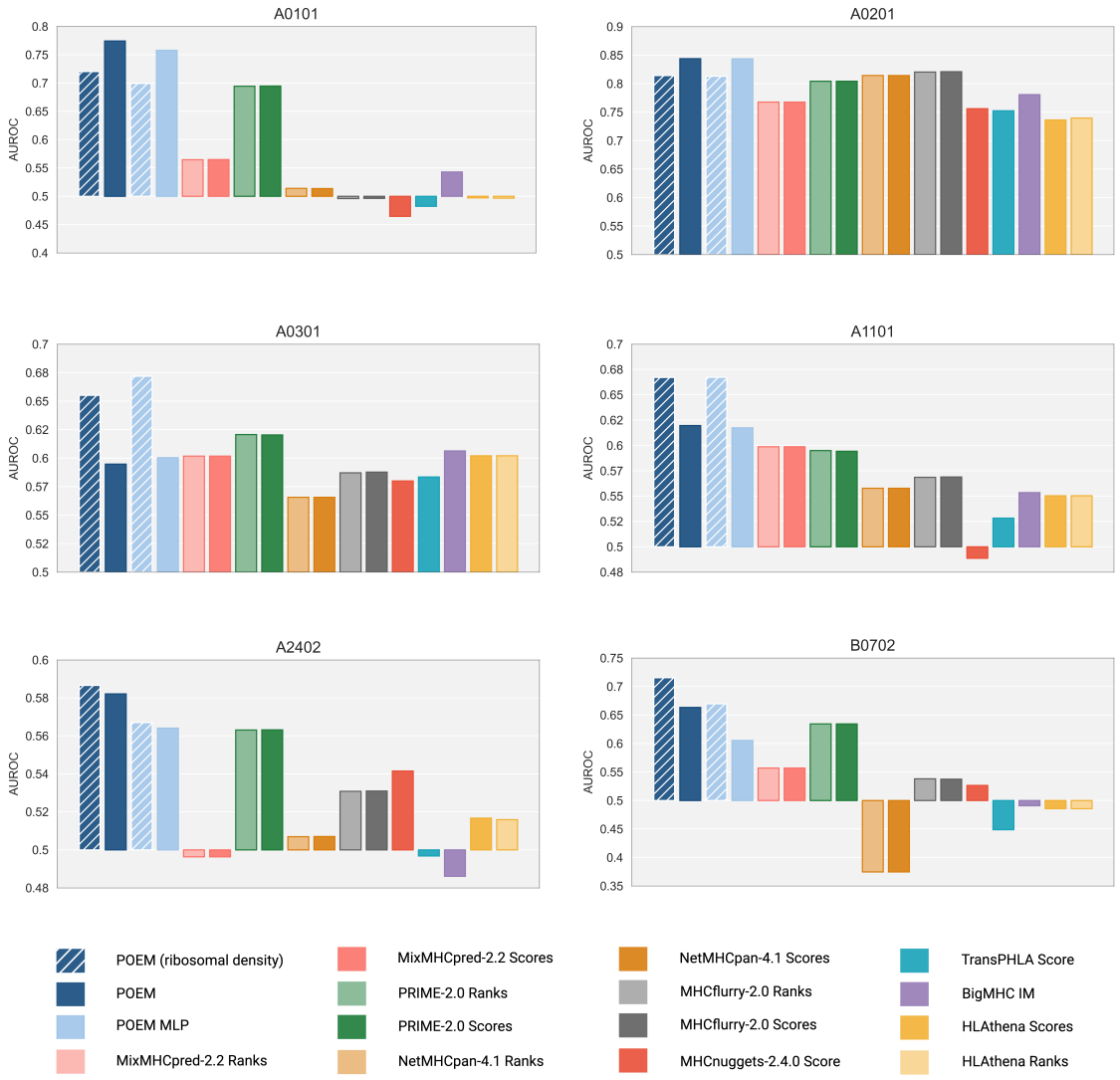


Figure 7.11. POEM AUROC compared against other prominent immunogenicity predictors for a dataset of SARS-CoV-2 epitopes with experimentally validated immunogenicity status across the 6 most abundant alleles in IEDB.

7.3.2.5 Source protein expression can further enhance POEM predictions

As described in Section 7.2.1.4, we assumed a homogeneous supply of 1 copy of the source protein per second for each candidate epitope. However, this assumption of homogeneity is likely to be inaccurate in many situations because proteins are not equally expressed.

In the case of SARS-CoV-2, vastly different ribosomal densities have been reported across the various open reading frames (ORFs), suggesting differences in the supply of these proteins available for processing by the proteasome [58]. We used these reported ribosomal densities to further scale the predicted protein supply, retaining a supply of 1 protein per second for the ORF with the highest ribosomal density

(nucleocapsid) but re-scaling the supply of other ORFs by their ribosomal density relative to nucleocapsid.

We found that the inclusion of this ribosomal density data had a positive effect on the AUROC for 4 of the 6 alleles (denoted by the striped bars in Figure 7.11). In particular, for A0301, the inclusion of ribosomal density data increased the AUROC from 0.59 to 0.66, improving performance to beyond that of PRIME-2.0. However, the inclusion of ribosomal density data also led to a drop in AUROC for HLA-A*01:01 and HLA-A*02:01, although only to a level around or above the next best-performing methods.

7.3.3 Sources of POEM efficacy

We conclude the results section of this chapter by attempting to better understand the sources of POEM's efficacy. The most direct comparison can be made to *PRIME-2.0*, which is trained on the same dataset and has a similar composition of input features but appears to have significantly lower accuracy than POEM across the majority of test cases considered. Hence, our aim was to understand which of the differences between POEM and *PRIME* were the source of this efficacy.

7.3.3.1 Mechanistic model predictions

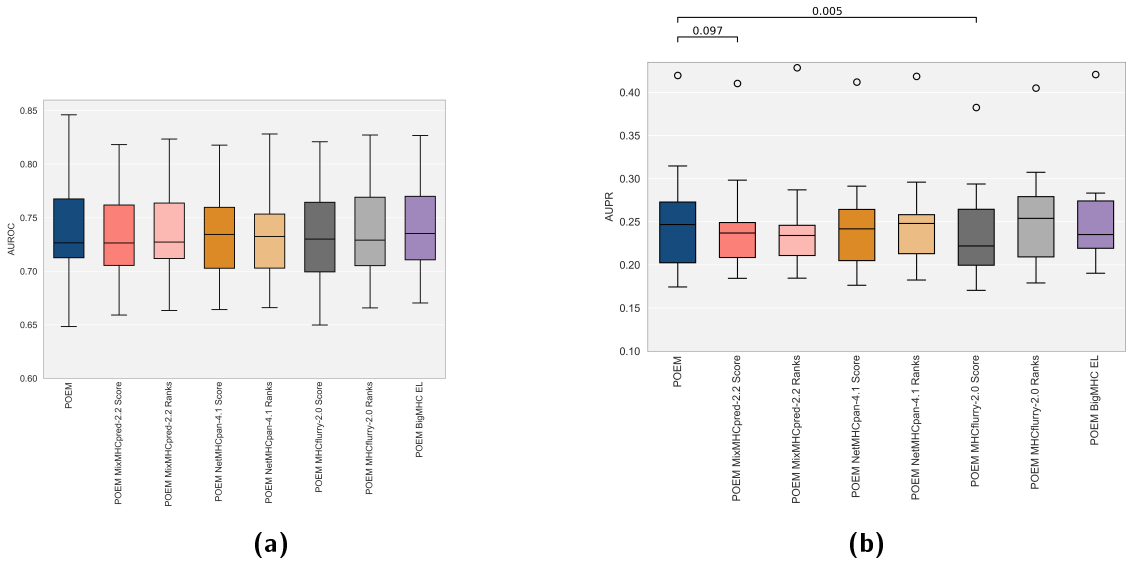
To test the extent to which the mechanistic model of antigen processing and presentation developed in this thesis represents an improvement over existing machine learning algorithms predicting antigen presentation, we re-trained POEM, replacing the mechanistic prediction with predicted scores or ranks from prominent predictors of antigen presentation from the literature.

For the *PRIME-2.0* training data, we did not find a significant difference between the AUROC using mechanism and the AUROC using any of the machine learning algorithms (Figure 7.12a). When considering precision-recall in Figure 7.12b, we found a significant drop in AUPR when mechanism was replaced with the MixMHCpred-2.2 score or the MHCflurry-2.0 score (although not when using the ranks returned by these models).

We then tested the performance of these models on the other test sets and were able to observe more substantial differences in model behaviour (Figure 7.12). On the GBM dataset, the mechanistic implementation resulted in a far superior performance than its machine learning counterparts when assessed using AUROC or AUPR. On the IEDB dataset (shown in Figure A.5), we did not find much difference across

the models, with the *NetMHCpan* methods and *MixMHCpred-2.2* scores performing slightly better than the rest of the methods.

Across the SARS-CoV-2 dataset, the differences between the mechanistic model and the alternatives were less pronounced. However, the use of our mechanistic model led to the best performance in terms of AUROC on 4 of the 6 alleles (A0101, A0201, A1101 and A2402), albeit only by a slim margin. The mechanistic version of POEM performed better than both *NetMHCpan-4.1* versions across all 6 alleles, both *MHCflurry-2.0* versions on 5 of the 6 alleles, and *BigMHC EL* on 4 of the 6 alleles. When compared to the *MixMHCpred-2.2* ranks (used by *PRIME-2.0*), the mechanistic approach was superior on 4 of the 6 alleles. Similar relative performance was seen when comparing AUPR (in Figure A.5). We concluded from this that the mechanistic input has a significant (generally positive) impact on POEM's predictions, but is not solely responsible for the increase in efficacy seen relative to *PRIME-2.0* and other methods.



(a) **(b)**

Figure 7.12. Comparison of POEM performance on *PRIME-2.0* dataset with POEM retrained replacing mechanistic model prediction with machine learning algorithm predictions of pMHC abundance. No significant difference was found between the AUROC (a) of POEM and the other methods. The p-values on the AUPR boxplot in (b) indicate the result of a one-tailed paired Wilcoxon signed rank test between POEM and the other methods, with only significant results shown.

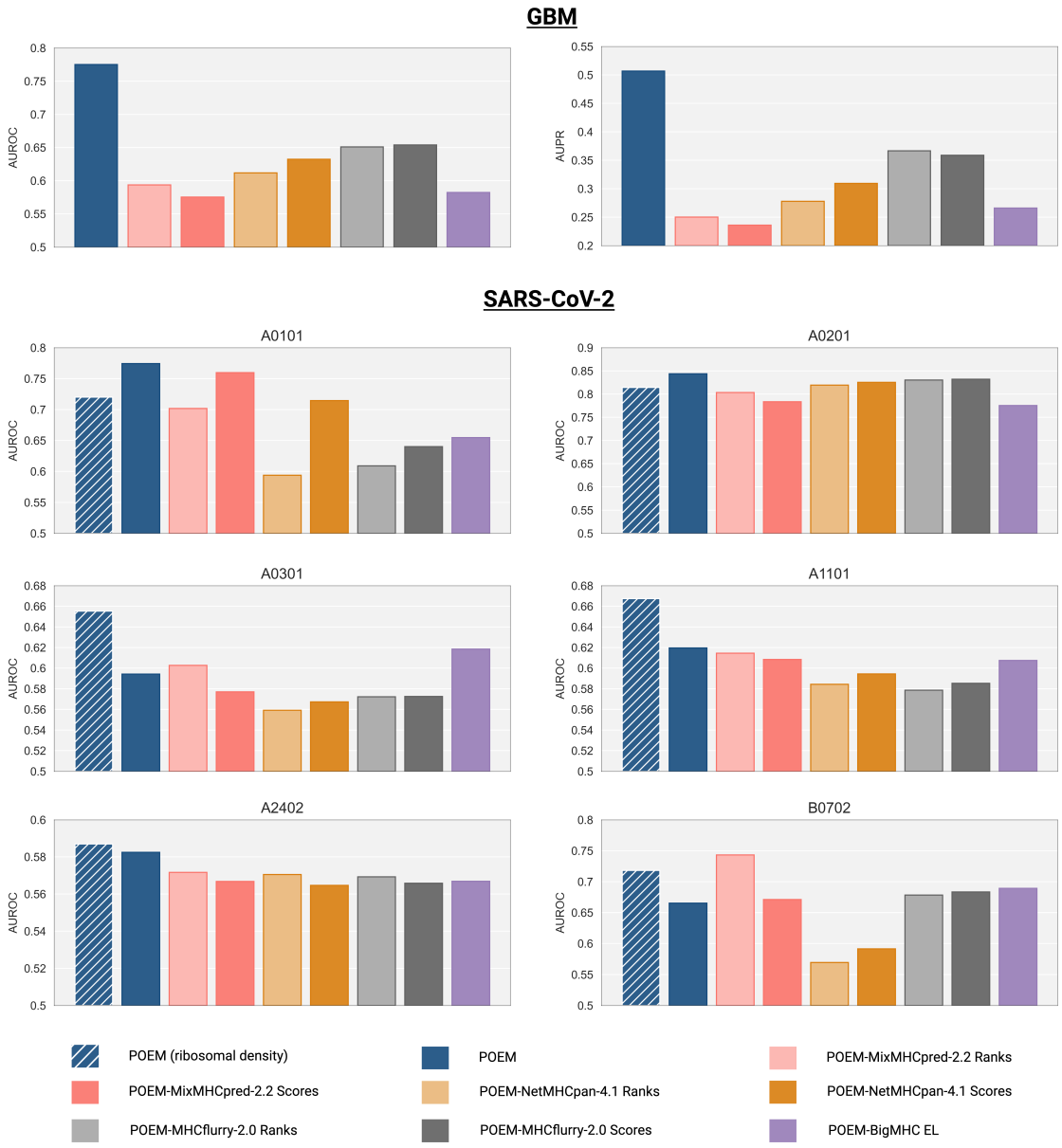


Figure 7.13. Comparison of POEM performance on test datasets with POEM re-trained replacing mechanistic model prediction with machine learning algorithm predictions of pMHC abundance.

7.3.3.2 POEM ablations

To understand the relative contribution of the 4 types of input to POEM, we performed an ablation study. We retrained the model from scratch without the inclusion of different features and measured the effect on performance via the 10-fold cross-validation on the training set.

We found that only the removal of the MHC-I allele and the mechanistic prediction resulted in a significant drop in performance at the $p \leq 0.01$ level (Figure 7.14),

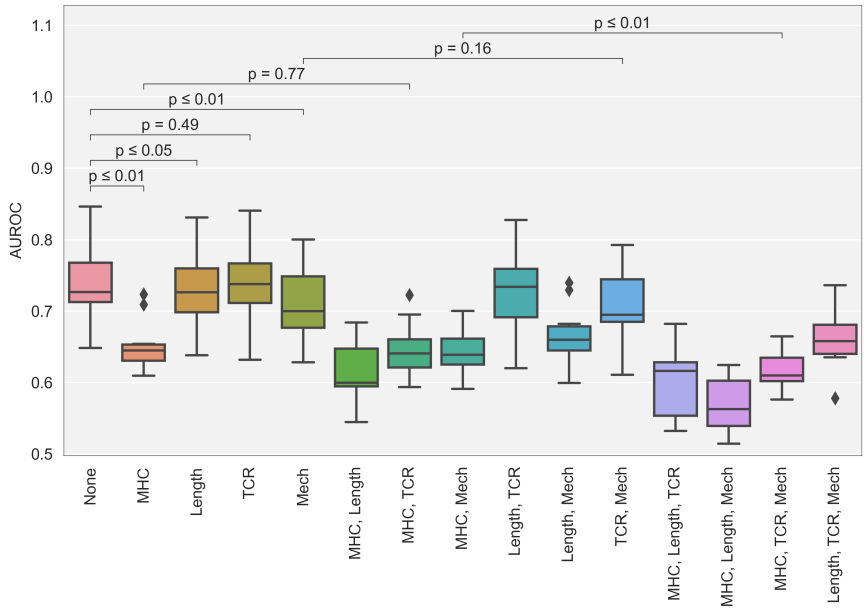


Figure 7.14. Comparison of cross-validation performance of POEM following ablation of features/combinations of features. p-values correspond to the results of a two-tailed paired Wilcoxon signed rank test.

although the removal of the length encoding was also significantly detrimental at the 0.05 threshold. The TCR encoding appeared to have the smallest effect on POEM's performance. Exclusion of this input feature alone resulted in a modest increase in AUROC, although not significant ($p = 0.49$). Furthermore, exclusion of TCR in combination with other features did not generally result in a significant drop in AUROC beyond what was observed with the exclusion of the other features alone. The only exception to this was the exclusion of the TCR encoding in addition to the MHC-I allele and mechanistic prediction which resulted in a significant fall in AUROC beyond what was observed with only the MHC-I and mechanistic ablation alone ($p \leq 0.01$).

7.3.3.3 Feature importance

Finally, to directly see the importance being placed by POEM upon each input feature, we examined the feature importance of the logistic regressor by means of comparison of the magnitudes of the coefficients (shown in Figure 7.15). We found that the feature with the highest contribution to the immunogenicity score was the BLOSUM50 representation of position 97 in the MHC-I allele (weight -3.40). This was closely followed by the mechanistic model output (3.31), and MHC-I residues 62 and 114 (-2.58 and -2.37 respectively). The TCR-interacting peptide residues and the length encoding were found to have a substantially lower feature importance, with only the

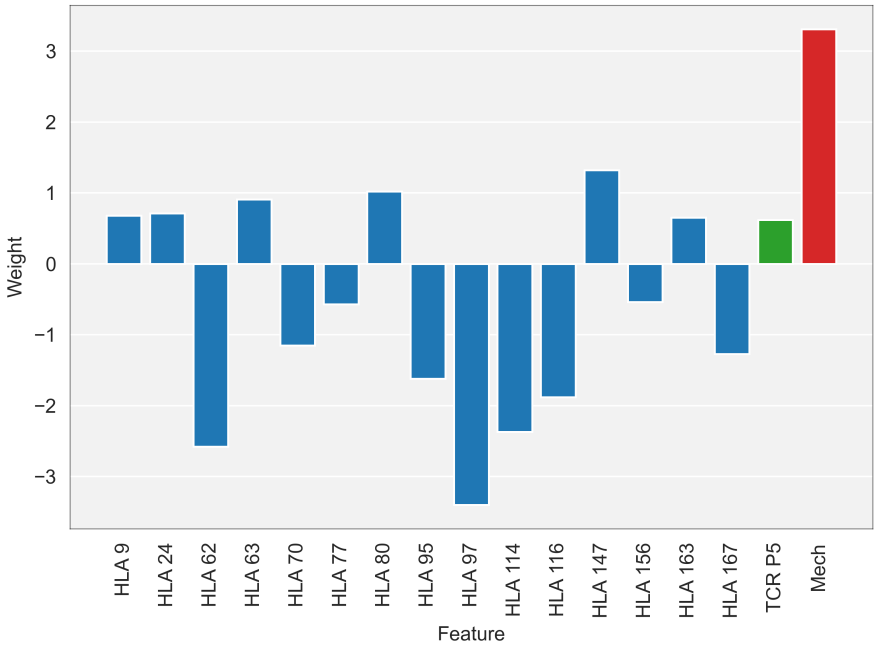


Figure 7.15. Comparison of logistic regression coefficients corresponding to input features. Only features with an absolute coefficient weight of 0.5 or more are shown. Colour denotes the category of input feature.

encoding corresponding to site 5 in the peptide (P5) having a coefficient above 0.50. This is broadly consistent with the trend seen in the ablation study.

7.4 Discussion

7.4.1 Efficacy of POEM

Across the training data and the test sets considered, POEM appears to perform extremely competitively with state-of-the-art methods. This could be suggestive of improved accuracy in the prediction of antigen presentation resulting from the use of the mechanistic model developed in this thesis. This notion is supported by the finding that replacing the mechanistic prediction with other machine learning algorithms led to a sizeable drop in performance on the GBM dataset and certain alleles in the SARS-CoV-2 dataset.

We found that training an MLP did not lead to a significant increase in accuracy on the training set (a result also observed by Gfeller et al. in their development of *PRIME-2.0*) but did appear to enhance performance on the GBM dataset relative to the logistic regressor. However, the MLP's performance on A2402 and B0702 in the SARS-CoV-2 dataset was notably worse than the logistic regressor. This is suggestive of possible overfitting occurring, since the GBM dataset is all associated with A0201 — the most prevalent allele in the training dataset by far. This serves as a reminder that higher dimensional models may generalise poorly to unseen data, which may explain why POEM — a simple logistic regressor — is able to perform better than more complex machine learning techniques (e.g. long short-term memory networks) on some of these test datasets. The strong performance of all models observed in the prediction of A0201 restricted COVID epitopes further supports this assessment, since this is the most common allele in publicly available training data.

Despite being trained on exclusively neoantigen data, POEM appears to be able to accurately identify immunogenic epitopes in a pathogenic context. This suggests that similar rules exist underpinning immunogenicity for both neoantigens and pathogenic antigens, which is consistent with our immunological understanding of how a cellular immune response is initiated.

7.4.2 Limitations

7.4.2.1 TCR frequency

The development of POEM was focused upon characterising antigen presentation and recognition by a TCR. We include no prediction of cognate TCR supply in our model inputs, so are currently implicitly assuming that a homogeneous supply of

cognate TCR for any presented pMHC complex will be available. This is likely to be highly inaccurate, particularly for the purpose of neoantigen prediction.

The repertoire of TCRs is the product of thymic selection, during which TCRs binding to self antigens are removed. As a result, antigens with high sequence similarity to endogenous peptides are less likely to have a cognate TCR present after this process. Certain predictors of CD8+ epitope immunogenicity include a measure of antigen similarity to self peptides in their scoring function, thus penalising these self-similar peptides. This was beyond the scope of this thesis but the inclusion of such a score may well augment our predictions, especially for neoantigens, which typically have a higher sequence similarity to the endogenous peptide cargo.

7.4.2.2 Binding affinity prediction

In order to use our mechanistic model, we require off-rates to be predicted for the epitope and its precursors. To do so, we rely on *NetMHC-4.0* (and, in some cases, *NetMHCpan-4.1*). Our analysis in Chapter 6 revealed the high sensitivity of the mechanistic model to these off-rate parameters, thus highlighting the importance of accurate predictions. However, comparative tests of these algorithms against experimentally measured binding affinities have revealed weak correlations for strong binders [173]. Hence, the use of these algorithms is currently introducing an uncertainty into highly important parameters, affecting the reliability of its predictions. Where feasible, off-rates could be measured accurately using techniques like BFA decay assays. However, this would greatly reduce the throughput of POEM and restrict its value as an *in silico* screening tool.

7.4.2.3 Rare HLA alleles

For rarer HLA alleles, Bashirova et al. did not measure tapasin dependence scores. We have already shown in Chapter 5 that this is a challenging property to predict for new alleles but has a significant effect on the composition and quantity of pMHC complexes on the presenting cell surface. Hence, this is currently a limitation in our model that may lead to inaccurate predictions for rarer alleles.

7.4.2.4 Computation time

To run POEM takes on the order of 1 second on a standard laptop. This means that many thousands of predictions can be formed per hour and hundreds of thousands per day. However, many of the methods presented in Table 7.1 can form predictions

in under 10% of this time, providing a small advantage in their use as high throughput screening tools.

7.4.3 Future directions

We found that the incorporation of an estimate of source protein translation rate using ribosomal density was able to enhance POEM performance for certain alleles on the SARS-CoV-2 dataset. This observation is consistent with other recent advances in the field, in which several studies have noted that the incorporation of protein expression data, or even transcriptomics data, appears to enhance epitope discovery [63, 87, 162]. Accordingly, we intend to develop this into future versions of POEM. We would expect the ribosomal density data to give us a more accurate estimate than transcriptomics data of the supply of protein for degradation by the proteasome, under the assumption that the supply of newly translated proteins will be offset by degradation in order to maintain a steady-state abundance. However, public transcriptomics datasets are much more prevalent in the literature, so it may be necessary to use these in order to maximise our training dataset.

We also currently only simulate the processing and presentation of a peptide by a single HLA allotype at a time. In reality, humans express 2 different HLA-A, -B, and -C alleles, simultaneously loading and presenting peptides on the cell surface. For alleles with overlapping binding motifs, this competition may have a detrimental effect on the presentation and immunogenicity of certain peptides. Hence, we intend to adapt our model to simultaneously include different MHC-I alleles and more faithfully represent the underlying immunology.

7.4.4 Concluding remarks

In this chapter we have used the mechanistic model of antigen processing and presentation developed in thesis to train a novel predictor of CD8+ immunogenicity. We call the resulting model POEM and have demonstrated impressive efficacy that appears to be superior in many places to the current state-of-the-art. Although not entirely, the use of mechanistic rather than machine learning predictions of pMHC presentation appears to contribute to POEM's efficacy. The other main source of improved efficacy is the inclusion of the MHC-I pseudosequence.

We intend to continue to develop POEM and add addditional features, which we anticipate should contribute further to its efficacy. This should hopefully render POEM

a valuable tool which can support the epitope prediction process for (personalised) cancer immunotherapy development, amongst other purposes.

Chapter 8

Discussion

8.1 Research goals

Having presented the research conducted over the course of this DPhil, I shall conclude by evaluating the extent to which we achieved our research goals. The objectives of this DPhil set out in Chapter 1 were twofold:

1. To develop and validate a mechanistic model of the antigen processing pathway, integrating predictions of novel and existing machine learning models.

2. To use predicted pMHC abundance from the resulting model to train a new predictor of CD8+ immunogenicity.

I will now summarise chapter-by-chapter the main findings of our work and link them to these two goals. I will also highlight current areas of weakness in our work and present future research plans, both to mitigate the impact of these weaknesses and to extend the impact of the work in this thesis.

8.2 Research summary

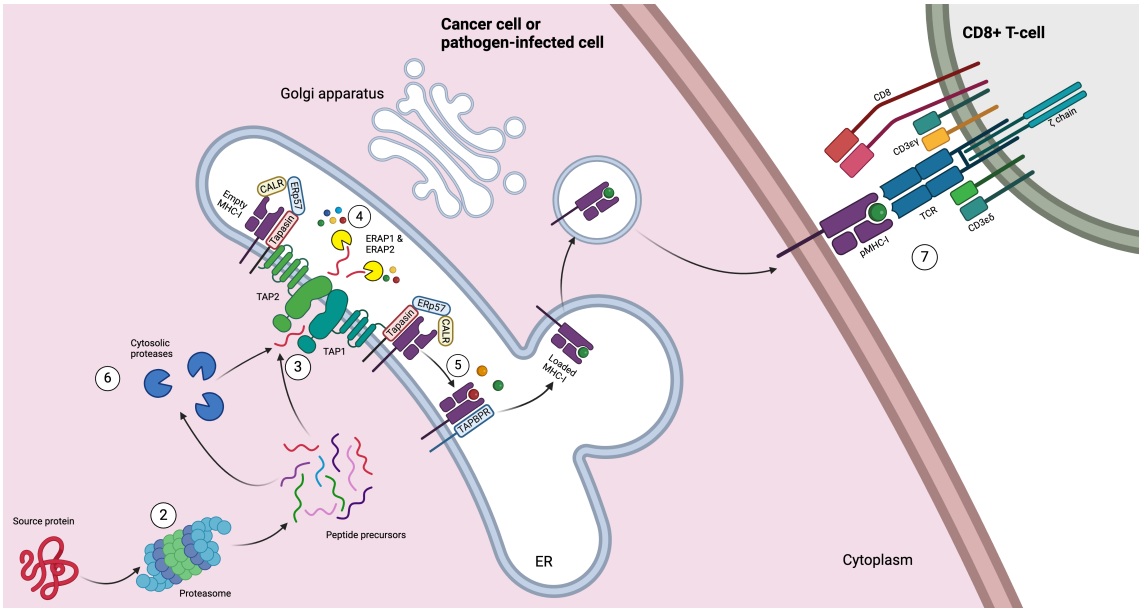


Figure 8.1. The schematic of the antigen processing pathway presented in Chapter 1 (Figure 1.3), with numbers indicating the chapter most applicable to that component.

In Chapter 2, we presented an adaptable framework for converting existing proteasomal cleavage algorithm predictions into predicted probabilities of specific products being formed by the proteasome. To align the model with proteasomal product length distributions reported in the literature, we found it necessary for the cleavage probability of each protein bond to depend on the cleavage status of preceding bonds. This introduces a directional dependence on how the protein is processed by the proteasome, with substantial differences in product formation probabilities, depending on whether the substrate enters from its N- or C-terminus. This result is consistent with observations in the literature of different proteasomal products identified when proteins were induced to enter the proteasome from either terminus [18]. We parametrised various scaling factors in our model for three families of cleavage algorithm in the literature: *NetChop*, *Pepsickle* and *BiLSTM*. We then evaluated the accuracy of our predictions of ovalbumin products using these algorithms, finding the strongest performance for members of the *Pepsickle* family of prediction algorithms.

In Chapter 3, we developed a new model of TAP binding affinity prediction called *PanTAP*, named for its use of training datasets from other mammals and its ability to form predictions for different species. In order to integrate data from binding affinity assays using mouse and rat TAP, we introduced a pseudosequence for TAP by using residues implicated in peptide binding from cryo-EM structures [99]. The notion of a pseudosequence to permit the training of a single binding model was inspired by

the same strategy's success in the *NetMHCpan* binding affinity predictor [114]. We employed a systematic approach to determine the optimal method of encoding amino acids and standardising peptide lengths for this problem, finding that an ensemble of different encodings resulted in a more predictive model. When compared to previous methods in the literature, *PanTAP* returned more accurate predictions of human TAP binding affinity. We concluded from this that the growth in training data from incorporating mouse and rat TAP assays had contributed additional information about the general rules of TAP–peptide binding. We anticipate that *PanTAP* will be a particularly useful tool for anyone wishing to predict binding affinities for non-human TAP (e.g. for studying pre-clinical class I antigen presentation).

In Chapter 4, we employed a similar approach to Chapter 3, this time to train a predictive model of ERAP1 trimming. We consolidated data from a particular type of enzymatic assay that was prevalent in the literature and trained a support vector regression (SVR) model to predict what the outcome would have been for other peptides, not studied in these assays. We then calibrated the output of these assays to Michaelis-Menten kinetics, thus enabling our SVR predictions to be used for enzyme kinetic parameter predictions for incorporation into mechanistic models.

We brought together the models of Chapters 2 to 4 and used them to extend Dalchau et al.'s systems biology model of peptide loading to MHC-I in Chapter 5. We re-parametrised the systems model using data from an H2-Kb transfected .220 cell line and considered how the resulting parameters might differ for different MHC-I alleles. We concluded that the peptide-MHC binding rate should allele-specific so that the model's predictions are consistent with the tapasin dependence of that allele. Although a large database of measured tapasin dependencies has been published [14], we wished to generalise this approach to all possible alleles, so in the remainder of Chapter 5 we investigated whether tapasin dependence could be predicted from MHC-I sequence. We concluded that tapasin dependence is a complicated property of the MHC-I allele that cannot be accurately predicted from sequence alone and likely requires molecular dynamics simulations to predict.

Our extended systems model from Chapter 5 was used to predict the antigen processing of N-terminally extended SIINFEKL in Chapter 6, further validating its predictions. The use of an equivalent dataset in an ERAP1 knockdown cell line also enabled us to parametrise the N-terminus trimming carried our by cytosolic aminopeptidases. The resulting parameter values appear to be consistent with a relevant *in vitro* assay in the literature, giving us confidence that these parameters hold more generally [139]. In Chapter 6 we also conduct a sensitivity analysis of our mechanistic model,

concluding that pMHC presentation sensitivity is dominated by peptide–MHC binding affinity and inferring a minor role for N-terminally extended precursors.

We were finally able to address the second of our objectives in Chapter 7, developing a predictor of CD8+ immunogenicity called *POEM*. The development of *POEM* was inspired by the design of *PRIME-2.0* and we used the same dataset to train our model [66]. In addition to using our mechanistic model's predictions, we found that including the restricting MHC-I allele's pseudosequence in the training features resulted in a significant improvement in predictive efficacy. We benchmarked *POEM* against prominent immunogenicity predictors from the literature, using a dataset of GBM neoantigens with validated immunogenicity, and finding that *POEM* performed substantially better than any of the other models.

We then tested *POEM*'s performance in a pathogenic context, finding superior performance on a dataset of pathogenic epitopes from IEDB, published by the authors of *BigMHC* [10]. To study a single pathogen in greater detail, we produced our own dataset of SARS-CoV-2 peptides from IEDB and used it to compare model predictions on the 6 most prevalent HLA alleles. Again, *POEM* showed impressive efficacy when compared to the other algorithms. We also found that using measured ribosomal density to scale source protein availability in our proteasome model resulted in a dramatic increase in performance on certain alleles.

8.3 Limitations

On the whole, we were able to address both of our original research goals across these chapters. However, in some areas our ability to do this was limited by various constraints.

8.3.1 Training data availability

A common limitation in mathematical modelling is the availability of appropriate data for model training. This was problematic for us in a few areas of the mechanistic model development, particularly in the training of the ERAP1 predictive model.

Our predictor of ERAP1 specificity was trained using only 85 unique peptides, of which 45 contain the sequence SIINFEKL. With the large combinatorial space of possible substrates (20^n for an n-mer) and ERAP1's specificity dependence on most residues in its substrate, this dataset is likely to be far too small to train an accurate predictor of ERAP1 trimming. Furthermore, our cross-validation assessment of

model performance is likely to be inflated by the high sequence similarity of the peptides in the training set, giving an inflated impression of model generalisation ability. I discuss an approach for inferring ERAP1 activity from immunopeptidomics data in Section 8.4.2 which could provide a means of training a new predictor without the need to carry out additional *in vitro* ERAP1 digestion assays.

8.3.2 Tapasin dependence

In order to extend our mechanistic model to the full breadth of possible HLA allotypes, we must either predict or measure the tapasin dependence of any alleles not included in the Bashirova dataset [14]. This is required to determine the appropriate peptide–MHC binding rate parameter to use. Prediction of this property appears to be non-trivial, as we found in Chapter 5, so this will likely require further experimental data to be generated using the same protocol as Bashirova et al..

8.3.3 ERAP2 omission

In my description of the antigen processing pathway from Chapter 1, I discussed ERAP2 and its role in the generation of class I epitopes, which is still in the process of being fully understood. We were unable to include ERAP2 in our model because of a lack of data available to train a predictor of its specificity. Furthermore, the experiment from Hearn et al. used to validate our mechanistic model was carried out in a HeLa cell line which did not express ERAP2 [71], so our mechanistic model did not need to include it to generate predictions consistent with their findings.

However, in a previous study, Hearn et al. compared the processing of ER-targeted SIINFEKL precursors in the same cell line with processing in an ERAP2-expressing cell line (COS 7) [72]. Although they found a strong correlation between the two ($R^2 = 0.7403$), certain residues, including leucine, were seemingly removed less efficiently in the presence of ERAP2, showing qualitative agreement with other reports of ERAP2 activity [137]. These differences could be more pronounced for different peptide sequences and lengths, and association of ERAP2 with various diseases has been reported, further implying an important role in antigen processing. Hence, we may wish to revise our omission of ERAP2 as its role in the pathway is better understood.

8.3.4 Homogeneous validation datasets

In many areas of our mechanistic model development, we used training or validation datasets containing peptides with high sequence similarity. In particular, the canonical epitope SIINFEKL and variants of this peptide were used in:

- Validation of the proteasome model's predictions (Chapter 2).
- Training of the ERAP1 predictive model (Chapter 4).
- Parametrisation of the extended *Dalchau* model (Chapter 5).
- Validation of the mechanistic model using the Hearn et al. study (Chapter 6).

Our mechanistic model has therefore been carefully designed for its ability to accurately predict parameters associated with this peptide at each stage of its development. Furthermore, the same MHC-I allele (H2-Kb) was used in both the Dalchau et al. study and the Hearn et al. study, albeit in different transfected cell lines (.220 and HeLa cells). This is concerning because our parameter prediction may be overfitting on SIINFEKL and the mechanistic parameters (e.g. MHC-I supply rate) may not be applicable to other MHC-I alleles. To investigate whether this is a problem, we would ideally use a similar study to Hearn et al.'s study, carried out using a peptides with highly dissimilar sequences to SIINFEKL. However, no such study could be found in the literature, restricting our ability to test the mechanistic model's generalisation performance.

8.3.5 POEM performance

Finally, despite better performance, POEM is slower to run than its machine learning counterparts. Many stages are involved in predicting the immunogenicity of a single 9mer:

1. Identify source protein (e.g. using pBLAST on reference proteome).
2. Run *Pepsickle* on the source protein sequence and use output to calculate probability of formation of 9mer and associated peptides (9mer to 16mer).
3. Run *NetMHCpan-4.1*, *PanTAP*, and ERAP1 prediction on these peptides.
4. Lookup cytosolic aminopeptidase trimming rates for these peptides and estimate the peptide–MHC binding rate from the tapasin dependence.
5. Simulate the mechanistic model to equilibrium.

6. Use the mechanistic model output to run *POEM*.

Although this pipeline is fully automated and can be parallelised if multiple peptides are queried simultaneously, the time per query is typically on the order of a few seconds. This is at least an order of magnitude slower than the majority of machine learning algorithms for predicting immunogenicity, so limits efficiency with which large databases of peptides can be queried.

We would therefore advocate for an initial filter to be applied if using *POEM* to systematically search an entire proteome (e.g. the SARS-CoV-2 proteome) for immunogenic peptides. A threshold predicted binding affinity would be a sensible means of reducing the size of the search space without requiring significant computation. A sensible value for this threshold could be estimated from predicted binding affinities of known immunogenic epitopes in order to keep the false negative (i.e. number of discarded immunogenic peptides) acceptably low.

8.4 Future work

8.4.1 Integration of protein expression data

When testing *POEM*'s predictions on the SARS-CoV-2 dataset, we found that the use of ribosomal density data to scale source protein supply to the proteasome seemed to improve predictive efficacy. This finding echoes recent observations in the literature, in which protein expression data has been used in tandem with MHC-I binding affinity predictions, leading to improved immunogenicity prediction [16, 30, 63, 87, 136, 162]. One of the easiest ways in which *POEM* could be improved would therefore be to incorporate this data into its training. The easiest way to do this would be to use the Peptide eXpression annotator (PepX) — a publicly available tool, using RNAseq data to estimate expression of a peptide's source protein [60]. It should be noted that RNAseq is not necessarily the most appropriate measurement for our mechanistic model, since the translational efficiency of the mRNA ultimately determines the protein availability. Ribosomal density data, on the other hand, directly measures translation, providing a more accurate measure of gene expression at the protein level. However, RNAseq data is far more prevalent, so for the next iteration of *POEM* it would be the most tractable way to incorporate a measure of protein expression into our analysis.

8.4.2 Reducing bias in machine learning

We have discussed how using epitope data to train a proteasomal cleavage predictor is confounded by the influence of other components of the antigen processing pathway (e.g. TAP and MHC-I binding affinities). However, we could use our mechanistic model to predict the impact of these stages on antigen presentation and thus potentially train a less biased cleavage predictor.

How this might work is shown in Figure 8.2. Because the system of ODEs is slow to simulate thousands of times, we should first train a simple neural network to emulate the predictions of our mechanistic model in a fraction of the time. To train this model proxy, we would need to simulate the mechanistic model a large number of times and for a wide range of randomly sampled parameter values, spanning the domain of possible parameters. We could then train a neural network to learn the mappings from each set of parameters to the mechanistic model outputs (pMHC abundance). With this model proxy trained, we could then efficiently train a cleavage predictor, using a similar approach to *NetChop* or *Pepsickle*. At each iteration of neural network fitting, we would predict the cleavage probabilities for the source proteins of the epitopes in our training set. These would then be fed into the model proxy, rapidly predicting the pMHC abundance of the epitopes and random negative decoy peptides, given these predicted cleavage probabilities. The predicted pMHC abundances would then be used to evaluate the loss function along with the training dataset labels.

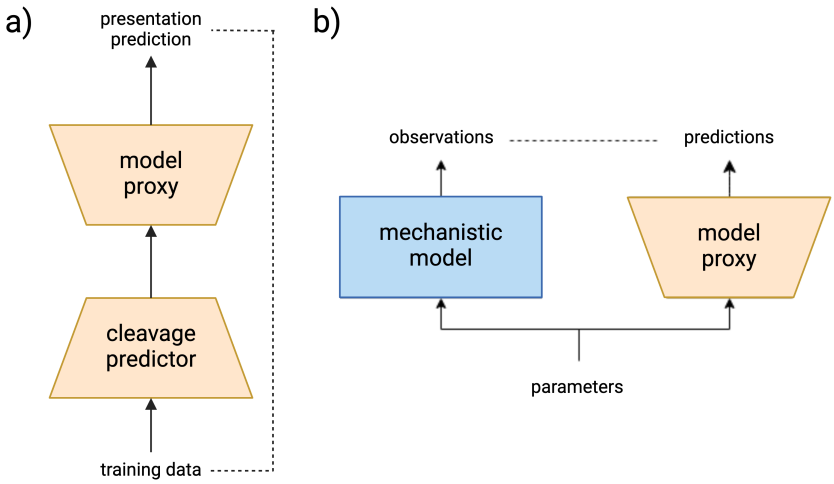


Figure 8.2. Schematic representation of the proposed method. Arrows represent flow of information. Dashed lines indicate quantities used for loss calculation. Neural networks are represented by orange trapezia, the mechanistic model by a blue rectangle. a) Training of a predictor of proteasomal cleavage using pre-trained proxy model to emulate mechanistic model predictions. b) Training of a proxy model to emulate mechanistic model predictions for different proteasomal cleavage inputs. Diagram inspired by [21].

A similar framework could be employed to train a predictor of ERAP1 trimming. Instead of using a dataset of epitopes, we could compare the (immuno-)peptidomes of wild-type cells with (immuno-)peptidomes of ERAP1 knockout/knockdown cell, potentially using datasets from Grey Wolf Therapeutics [164].

Hence, our mechanistic model may be used to train machine learning models to predict parameters for different stages of the antigen processing pathway by predicting the effect on the observed pMHC presence or abundance.

8.4.3 Tumour evolution and immune escape

Our mechanistic model could be applied to existing longitudinal studies of tumour evolution to simulate how changes in antigen presentation due to tumour evolution can contribute to immune escape. This leverages one of the main advantages of mechanistic modelling over machine learning discussed in Chapter 1: the ability to simulate changes in intracellular conditions by changing specific parameter values. For example, we could simulate the effect of loss of heterozygosity in the MHC-I genes by removing the corresponding MHC-I molecules from the ER compartment in our systems model. This would not be possible to do using existing machine learning methods. One could not change the weights in a pre-trained neural network (e.g. *MHCflurry-2.0*) because the weights do not correspond to physical entities in the underlying immunology (e.g. MHC-I supply).

A suitable dataset for this type of experiment would be the TracerX study [7]. Al Bakir et al. report a longitudinal analysis of 126 non-small cell lung cancer tumours from patients who developed metastatic disease, compared with a control cohort of 144 non-metastatic tumours. By comparing simulations of antigen presentation between the two cohorts, we would hope to explain the differences in disease progression in the context of tumour escape of immunosurveillance.

8.4.4 Future pandemics

Finally, POEM may be used to accelerate and derisk vaccine development for the next pandemic. There are many potential candidates for what this might be, and COVID-19 served as a warning shot, highlighting the ease with which a virus can spread in a globalised and heavily populated world. Vaccine development for SARS-CoV-2 was assisted by the spike protein containing many immunogenic epitopes, with reasonable coverage across HLA allotypes. However, we may not be so fortunate with the next pandemic and may need to rapidly predict immunogenic epitopes from

Predictor	H2-Kb		H2-Db	
	R_p	R_s	R_p	R_s
BigMHC EL [10]	-0.148	0.479	-0.194	-0.327
NetMHCpan-4.1 [130]	0.249	0.297	0.542	0.009
MHCflurry-2.0 [118]	0.367	0.527	0.220	-0.227
Mechanistic model	**0.613**	**0.564**	**0.668**	**0.445**

Table 8.1. Correlation between outputs machine learning and mechanistic predictors of antigen presentation, and the absolute quantification of direct presentation from the Wu et al. dataset. Performance is separated by restricting MHC-I allele. Correlation is given in terms of the Pearson correlation coefficient, R_p, and the Spearman correlation coefficient, R_s.

the pathogen's sequencing data. POEM could be an invaluable tool for this process as our study into SARS-CoV-2 showed that POEM is better performing than existing immunogenicity predictors, particularly on the less common HLA allotypes.

8.5 Closing remarks

In Chapter 1, to illustrate the problem with treating antigen presentation as a binary classification problem, I used a dataset of quantified pMHC abundance of identified influenza A peptides in a dendritic cell line [168]. I demonstrated the poor performance of well-known predictors of antigen presentation on this dataset, and proposed that this was likely to be the result of using these binary eluted ligand training sets. Having developed our mechanistic model of antigen presentation to address this, it feels appropriate to conclude this thesis by testing whether our mechanistic approach results in improved predictive accuracy across this dataset. I used *NetMHCpan-4.1* to predict the binding affinities of the epitopes and their precursors and simulated our mechanistic model to steady state. Since a tapasin dependence measurement for H2-Db could not be found in the literature, I set the peptide–MHC binding rate for H2-Db to the same as for H2-Kb. The mechanistic model led to an improvement over the machine learning approaches, both in terms of ranking of epitopes (using Spearman's correlation coefficient, R_s) and in terms of linear correlation between predicted and observed presentation (using Pearson's correlation coefficient, R_p), as shown in Table 8.1.

Although the Wu et al. dataset is too small and homogeneous to draw significant conclusions from, it is encouraging to see that our mechanistic model appears to

go some way towards solving the motivating problem from Chapter 1. With this type of direct evidence of more accurate pMHC abundance prediction, along with improved immunogenicity prediction through POEM, we can make a strong case for the future of mechanistic modelling in a world and field currently dominated by machine learning.

Appendix A

Appendix

A.1 Extended Dalchau model

The set of ordinary differential equations defining the extended *Dalchau* model presented in Chapter 5 are given by:

$$\frac{\mathrm{d}P_i^C}{\mathrm{d}t} = g_i + u^S \cdot SP_i^C - \left(d_C + b_i^S \cdot S\right) \cdot P_i^C \tag{A.1}$$

$$\frac{\mathrm{d}P_{self}^C}{\mathrm{d}t} = g_{self} + u^S \cdot SP_{self}^C - \left(d_C + b_{self}^S \cdot S\right) \cdot P_{self}^C \tag{A.2}$$

$$\begin{aligned} \frac{\mathrm{d}S}{\mathrm{d}t} &= \left(k^S + u^S\right)\left(SP_i^C + SP_{self}^C\right) \\ &\quad - S \cdot \left(b_i^S \cdot P_i^C + b_{self}^S \cdot P_{self}^C\right) \end{aligned} \tag{A.3}$$

$$\frac{\mathrm{d}SP_i^C}{\mathrm{d}t} = b_i^S \cdot P_i^C \cdot S - \left(k^S + u^S\right) \cdot SP_i^C \tag{A.4}$$

$$\frac{\mathrm{d}SP_{self}^C}{\mathrm{d}t} = b_{self}^S \cdot P_{self}^C \cdot S - \left(k^S + u^S\right) \cdot SP_{self}^C \tag{A.5}$$

$$\frac{\mathrm{d}P_b^{ER}}{\mathrm{d}t} = k^S \cdot SP_{self}^C \cdot \rho_b + u_b \cdot MP_b^{ER} + q \cdot u_b \cdot MP_b^{ER} - (c \cdot TM + b \cdot M + d_P) \cdot P_b^{ER} \tag{A.6}$$

$$\frac{\mathrm{d}P_n^{ER}}{\mathrm{d}t} = k^S \cdot SP_{self}^C \cdot \rho_n + u_n \cdot MP_n^{ER} + q \cdot u_b \cdot MP_n^{ER} - (c \cdot TM + b \cdot M + d_P) \cdot P_n^{ER} \tag{A.7}$$

$$\frac{\mathrm{d}P_i^{ER}}{\mathrm{d}t} = k^S \cdot SP_i^C + u_i MP_i^{ER} + q \cdot u_i \cdot MP_i^{ER} - \left[c \cdot TM + b \cdot M + d_P + \frac{k_i \cdot E_0}{\left(K_M + P_i^{ER} + P_n^{ER} + P_b^{ER}\right)} \right] \cdot P_i^{ER} \tag{A.8}$$

$$\frac{\mathrm{d}M}{\mathrm{d}t} = g_M + u_n \cdot MP_n + u_b \cdot MP_b + u_i \cdot MP_i + u_T \cdot TM - [b \cdot (P_b + P_n + P_i) + d_M + b_T \cdot T] \cdot M \tag{A.9}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = g_T + u_T \cdot TM + u_T \cdot v \cdot (TMP_b + TMP_n + TMP_i) - (b_T \cdot M + d_T) \cdot T \tag{A.10}$$

$$\frac{\mathrm{d}TM}{\mathrm{d}t} = b_T \cdot T \cdot M + q \cdot (u_n \cdot TMP_n + u_b \cdot TMP_b + u_i TMP_i) - [u_T + c \cdot (P_b + P_n + P_i)] \cdot TM \tag{A.11}$$

$$\frac{\mathrm{d}MP_b}{\mathrm{d}t} = b \cdot M \cdot P_b + u_T \cdot v \cdot TMP_b - (u_b + e)\, MP_b \tag{A.12}$$

$$\frac{\mathrm{d}MP_n}{\mathrm{d}t} = b \cdot M \cdot P_n + u_T \cdot v \cdot TMP_n - (u_n + e)\, MP_n \tag{A.13}$$

$$\frac{\mathrm{d}MP_i}{\mathrm{d}t} = b \cdot M \cdot P_i + u_T \cdot vTMP_i - (u_i + e)\, MP_i \quad \text{(A.14)}$$

$$\frac{\mathrm{d}TMP_n}{\mathrm{d}t} = c \cdot TM \cdot P_n - (q \cdot u_n + u_T \cdot v) \cdot TMP_n \quad \text{(A.15)}$$

$$\frac{\mathrm{d}TMP_b}{\mathrm{d}t} = c \cdot TM \cdot P_b - (q \cdot u_b + u_T \cdot v) \cdot TMP_b \quad \text{(A.16)}$$

$$\frac{\mathrm{d}TMP_i}{\mathrm{d}t} = c \cdot TM \cdot P_i - (q \cdot u_i + u_T \cdot v) \cdot TMP_i \quad \text{(A.17)}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_n - u_n \cdot M_eP_n \quad \text{(A.18)}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_b - u_b \cdot M_eP_b \quad \text{(A.19)}$$

$$\frac{\mathrm{d}M_eP_i}{\mathrm{d}t} = e \cdot MP_i - u_i \cdot M_eP_i \quad \text{(A.20)}$$

$$\frac{\mathrm{d}M_e}{\mathrm{d}t} = u_b \cdot M_eP_b + u_n \cdot M_eP_n + u_i \cdot M_eP_i - d_{M_e}M_e \quad \text{(A.21)}$$

where the subscript, i, is used to distinguish an exogenous peptide from the endogenous peptides, so denotes the SIINXEKX peptides in the context of the extended *Dalchau* model fitting.

A.2 Hearn et al. model

The set of ordinary differential equations used to simulate the Hearn et al. study in Chapter 6 is given by:

$$\frac{\mathrm{d}P^C_{XXSL}}{\mathrm{d}t} = g_{XXSL} + u^S \cdot SP^C_{XXSL} - \left(d_C + b^S_{XXSL} \cdot S + r_X\right) \cdot P^C_{XXSL} \quad \text{(A.22)}$$

$$\frac{\mathrm{d}P^C_{XSL}}{\mathrm{d}t} = r_X \cdot P_{XXSL} + u^S \cdot SP^C_{XSL} - \left(d_C + b^S_{XSL} \cdot S + r_X\right) \cdot P^C_{XSL} \quad \text{(A.23)}$$

$$\frac{\mathrm{d}P^C_{SL}}{\mathrm{d}t} = r_X \cdot P_{XSL} + u^S \cdot SP^C_{SL} - \left(d_C + b^S_{SL} \cdot S + r_S\right) \cdot P^C_{SL} \quad \text{(A.24)}$$

$$\frac{\mathrm{d}P^C_{self}}{\mathrm{d}t} = g_{self} + u^S \cdot SP^C_{self} - \left(d_C + b^S_{self} \cdot S\right) \cdot P^C_{self} \quad \text{(A.25)}$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \left(k^S + u^S\right)\left(SP^C_i + SP^C_{self}\right) \quad \text{(A.26)}$$
$$- S \cdot \left(b^S_{XXSL} \cdot P^C_{XXSL} + b^S_{XSL} \cdot P^C_{XSL} + b^S_{SL} \cdot P^C_{SL} + b^S_{self} \cdot P^C_{self}\right)$$

$$\frac{\mathrm{d}SP^C_{XXSL}}{\mathrm{d}t} = b^S_{XXSL} \cdot P^C_{XXSL} \cdot S - \left(k^S + u^S\right) \cdot SP^C_{XXSL} \quad \text{(A.27)}$$

$$\frac{\mathrm{d}SP^C_{XSL}}{\mathrm{d}t} = b^S_{XSL} \cdot P^C_{XSL} \cdot S - \left(k^S + u^S\right) \cdot SP^C_{XSL} \quad \text{(A.28)}$$

$$\frac{\mathrm{d}SP^C_{SL}}{\mathrm{d}t} = b^S_{SL} \cdot P^C_{SL} \cdot S - \left(k^S + u^S\right) \cdot SP^C_{SL} \quad \text{(A.29)}$$

$$\frac{\mathrm{d}SP^C_{self}}{\mathrm{d}t} = b^S_{self} \cdot P^C_{self} \cdot S - \left(k^S + u^S\right) \cdot SP^C_{self} \quad \text{(A.30)}$$

$$\frac{\mathrm{d}P^{ER}_b}{\mathrm{d}t} = k^S \cdot SP^C_{self} \cdot \rho_b + u_b \cdot MP^{ER}_b + q \cdot u_b \cdot MP^{ER}_b \quad \text{(A.31)}$$
$$- \left(c \cdot TM + b \cdot M + d_P\right) \cdot P^{ER}_b$$

$$\frac{\mathrm{d}P^{ER}_n}{\mathrm{d}t} = k^S \cdot SP^C_{self} \cdot \rho_n + u_n \cdot MP^{ER}_n + q \cdot u_b \cdot MP^{ER}_n \quad \text{(A.32)}$$
$$- \left(c \cdot TM + b \cdot M + d_P\right) \cdot P^{ER}_n$$

$$\frac{\mathrm{d}P^{ER}_{XXSL}}{\mathrm{d}t} = k^S \cdot SP^C_{XXSL} + u_{XXSL} MP^{ER}_{XXSL} + q \cdot u_{XXSL} \cdot MP^{ER}_{XXSL} \quad \text{(A.33)}$$
$$- c \cdot TM \cdot P^{ER}_{XXSL} + b \cdot M \cdot P^{ER}_{XXSL} + d_P \cdot P^{ER}_{XXSL}$$

$$- \frac{E_0 \cdot k_{XXSL} \cdot P^{ER}_{XXSL}}{\left(K_M + P^{ER}_{XXSL} + P^{ER}_{XSL} + P^{ER}_{SL} + P^{ER}_{n} + P^{ER}_{b}\right)}$$

$$\frac{\mathrm{d}P^{ER}_{XSL}}{\mathrm{d}t} = k^S \cdot SP^{C}_{XSL} + u_{XSL} MP^{ER}_{XSL} + q \cdot u_{XSL} \cdot MP^{ER}_{XSL} \tag{A.34}$$

$$- c \cdot TM \cdot P^{ER}_{XSL} + b \cdot M \cdot P^{ER}_{XSL} + d_P \cdot P^{ER}_{XSL}$$

$$+ \frac{E_0 \cdot \left(k_{XXSL} \cdot P^{ER}_{XXSL} - k_{XSL} \cdot P^{ER}_{XSL}\right)}{\left(K_M + P^{ER}_{XXSL} + P^{ER}_{XSL} + P^{ER}_{SL} + P^{ER}_{n} + P^{ER}_{b}\right)}$$

$$\frac{\mathrm{d}P^{ER}_{SL}}{\mathrm{d}t} = k^S \cdot SP^{C}_{SL} + u_{SL} MP^{ER}_{SL} + q \cdot u_{SL} \cdot MP^{ER}_{SL} \tag{A.35}$$

$$- c \cdot TM \cdot P^{ER}_{SL} + b \cdot M \cdot P^{ER}_{SL} + d_P \cdot P^{ER}_{SL}$$

$$+ \frac{E_0 \cdot \left(k_{XSL} \cdot P^{ER}_{XSL} - k_{SL} \cdot P^{ER}_{SL}\right)}{\left(K_M + P^{ER}_{XXSL} + P^{ER}_{XSL} + P^{ER}_{SL} + P^{ER}_{n} + P^{ER}_{b}\right)}$$

$$\frac{\mathrm{d}M}{\mathrm{d}t} = g_M + u_n \cdot MP_n + u_b \cdot MP_b + u_{XXSL} \cdot MP_{XXSL} \tag{A.36}$$

$$+ u_{XSL} \cdot MP_{XSL} + u_{SL} \cdot MP_{SL} + u_T \cdot TM$$

$$- \left[b \cdot (P_b + P_n + P_{XXSL} + P_{XSL} + P_{SL}) + d_M + b_T \cdot T\right] \cdot M$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = g_T + u_T \cdot TM + u_T \cdot v \cdot (TMP_b + TMP_n \tag{A.37}$$

$$+ TMP_{XXSL} + TMP_{XSL} + TMP_{SL}) - (b_T \cdot M + d_T) \cdot T$$

$$\frac{\mathrm{d}TM}{\mathrm{d}t} = b_T \cdot T \cdot M + q \cdot (u_n \cdot TMP_n + u_b \cdot TMP_b + u_{XXSL} TMP_{XXSL} \tag{A.38}$$

$$+ u_{XSL} TMP_{XSL} + u_{SL} TMP_{SL})$$

$$- \left[u_T + c \cdot (P_b + P_n + P_{XXSL} + P_{XSL} + P_{SL})\right] \cdot TM$$

$$\frac{\mathrm{d}MP_b}{\mathrm{d}t} = b \cdot M \cdot P_b + u_T \cdot v \cdot TMP_b - (u_b + e)\, MP_b \tag{A.39}$$

$$\frac{\mathrm{d}MP_n}{\mathrm{d}t} = b \cdot M \cdot P_n + u_T \cdot v \cdot TMP_n - (u_n + e)\, MP_n \tag{A.40}$$

$$\frac{\mathrm{d}MP_{XXSL}}{\mathrm{d}t} = b \cdot M \cdot P_{XXSL} + u_T \cdot vTMP_{XXSL} - (u_{XXSL} + e)\, MP_{XXSL} \tag{A.41}$$

$$\frac{\mathrm{d}MP_{XSL}}{\mathrm{d}t} = b \cdot M \cdot P_{XSL} + u_T \cdot vTMP_{XSL} - (u_{XSL} + e)\, MP_{XSL} \tag{A.42}$$

$$\frac{\mathrm{d}MP_{SL}}{\mathrm{d}t} = b \cdot M \cdot P_{SL} + u_T \cdot vTMP_{SL} - (u_{SL} + e)\, MP_{SL} \tag{A.43}$$

$$\frac{\mathrm{d}TMP_n}{\mathrm{d}t} = c \cdot TM \cdot P_n - (q \cdot u_n + u_T \cdot v) \cdot TMP_n \tag{A.44}$$

$$\frac{\mathrm{d}TMP_b}{\mathrm{d}t} = c \cdot TM \cdot P_b - (q \cdot u_b + u_T \cdot v) \cdot TMP_b \tag{A.45}$$

$$\frac{\mathrm{d}TMP_{XXSL}}{\mathrm{d}t} = c \cdot TM \cdot P_{XXSL} - (q \cdot u_{XXSL} + u_T \cdot v) \cdot TMP_{XXSL} \tag{A.46}$$

$$\frac{\mathrm{d}TMP_{XSL}}{\mathrm{d}t} = c \cdot TM \cdot P_{SL} - (q \cdot u_{XSL} + u_T \cdot v) \cdot TMP_{XSL} \tag{A.47}$$

$$\frac{\mathrm{d}TMP_{SL}}{\mathrm{d}t} = c \cdot TM \cdot P_{SL} - (q \cdot u_{SL} + u_T \cdot v) \cdot TMP_{SL} \tag{A.48}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_n - u_n \cdot M_eP_n \tag{A.49}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_b - u_b \cdot M_eP_b \tag{A.50}$$

$$\frac{\mathrm{d}M_eP_{XXSL}}{\mathrm{d}t} = e \cdot MP_{XXSL} - u_{XXSL} \cdot M_eP_{XXSL} \tag{A.51}$$

$$\frac{\mathrm{d}M_eP_{XSL}}{\mathrm{d}t} = e \cdot MP_{XSL} - u_{XSL} \cdot M_eP_{XSL} \tag{A.52}$$

$$\frac{\mathrm{d}M_eP_{SL}}{\mathrm{d}t} = e \cdot MP_{SL} - u_{SL} \cdot M_eP_{SL} \tag{A.53}$$

$$\begin{aligned}\frac{\mathrm{d}M_e}{\mathrm{d}t} = {} & u_b \cdot M_eP_b + u_n \cdot M_eP_n + u_{XXSL} \cdot M_eP_{XXSL} \\ & + u_{XSL} \cdot M_eP_{XSL} + u_{SL} \cdot M_eP_{SL} - d_{M_e}M_e\end{aligned} \tag{A.54}$$

where the subscripts $XXSL$, XSL and SL correspond to XXSIINFEKL, XSIINFEKL and SIINFEKL (and X represents an amino acid).

A.3 POEM mechanistic model

A.3.1 System of differential equations

The set of ordinary differential equations used to simulate the passage of epitopes and their precursors through the antigen processing pathway is given below.

$$\frac{\mathrm{d}P_i^C}{\mathrm{d}t} = r_{i+1} \cdot P_{i+1}^C + g_i + u^S \cdot SP_i^C - \left(d_C + r_i + b_i^S \cdot S\right) \cdot P_i^C \tag{A.55}$$

$$\frac{\mathrm{d}P_{self}^C}{\mathrm{d}t} = g_{self} + u^S \cdot SP_{self}^C - \left(d_C + b_{self}^S \cdot S\right) \cdot P_{self}^C \tag{A.56}$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \left(k^S + u^S\right) \sum_{i,self} SP_j^C - \sum_{i,self} b_i^S \cdot P_i^C \cdot S^C \tag{A.57}$$

$$\frac{\mathrm{d}SP_i^C}{\mathrm{d}t} = b_i^S \cdot P_i^C \cdot S^C - \left(k^S + u^S\right) \cdot SP_i^C \tag{A.58}$$

$$\frac{\mathrm{d}SP_{self}^C}{\mathrm{d}t} = b_{self}^S \cdot P_{self}^C \cdot S^C - \left(k^S + u^S\right) \cdot SP_{self}^C \tag{A.59}$$

$$\frac{\mathrm{d}P_b^{ER}}{\mathrm{d}t} = k^S \cdot SP_{self}^C \cdot \rho_b + u_b \cdot MP_b^{ER} + q \cdot u_b \cdot MP_b^{ER} - \left(c \cdot TM + b \cdot M + d_P\right) \cdot P_b^{ER} \tag{A.60}$$

$$\frac{\mathrm{d}P_n^{ER}}{\mathrm{d}t} = k^S \cdot SP_{self}^C \cdot \rho_n + u_n \cdot MP_n^{ER} + q \cdot u_b \cdot MP_n^{ER} - \left(c \cdot TM + b \cdot M + d_P\right) \cdot P_n^{ER} \tag{A.61}$$

$$\frac{\mathrm{d}P_i^{ER}}{\mathrm{d}t} = k^S \cdot SP_i^C + u_i MP_i^{ER} + q \cdot u_i \cdot MP_i^{ER} - \left(c \cdot TM + b \cdot M + d_P\right) \cdot P_i^{ER} + E_0 \cdot \left(k_{i+1} \cdot P_{i+1} - k_i \cdot P_i\right) / \left(K_M + \sum_i P_i^{ER} + P_n^{ER} + P_b^{ER}\right) \tag{A.62}$$

$$\frac{\mathrm{d}M}{\mathrm{d}t} = g_M + u_n \cdot MP_n + u_b \cdot MP_b + \sum_i u_i \cdot MP_i + u_T \cdot TM - \left[b \cdot \left(P_b + P_n + \sum_i P_i\right) + d_M + b_T \cdot T\right] \cdot M \tag{A.63}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = g_T + u_T \cdot TM + u_T \cdot v \cdot \left(TMP_b + TMP_n + \sum_i TMP_i\right) - \left(b_T \cdot M + d_T\right) \cdot T \tag{A.64}$$

$$\frac{\mathrm{d}TM}{\mathrm{d}t} = b_T \cdot T \cdot M + q \cdot \left(u_n \cdot TMP_n + u_b \cdot TMP_b + \sum_i u_i TMP_i\right) \tag{A.65}$$

$$- \left[u_T + c \cdot \left(P_b + P_n + \sum_i P_i \right) \right] \cdot TM$$

$$\frac{\mathrm{d}MP_b}{\mathrm{d}t} = b \cdot M \cdot P_b + u_T \cdot v \cdot TMP_b - (u_b + e)\, MP_b \tag{A.66}$$

$$\frac{\mathrm{d}MP_n}{\mathrm{d}t} = b \cdot M \cdot P_n + u_T \cdot v \cdot TMP_n - (u_n + e)\, MP_n \tag{A.67}$$

$$\frac{\mathrm{d}MP_i}{\mathrm{d}t} = b \cdot M \cdot P_i + u_T \cdot vTMP_i - (u_i + e)\, MP_i \tag{A.68}$$

$$\frac{\mathrm{d}TMP_n}{\mathrm{d}t} = c \cdot TM \cdot P_n - (q \cdot u_n + u_T \cdot v) \cdot TMP_n \tag{A.69}$$

$$\frac{\mathrm{d}TMP_b}{\mathrm{d}t} = c \cdot TM \cdot P_b - (q \cdot u_b + u_T \cdot v) \cdot TMP_b \tag{A.70}$$

$$\frac{\mathrm{d}TMP_i}{\mathrm{d}t} = c \cdot TM \cdot P_i - (q \cdot u_i + u_T \cdot v) \cdot TMP_i \tag{A.71}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_n - u_n \cdot M_eP_n \tag{A.72}$$

$$\frac{\mathrm{d}M_eP_n}{\mathrm{d}t} = e \cdot MP_b - u_b \cdot M_eP_b \tag{A.73}$$

$$\frac{\mathrm{d}M_eP_i}{\mathrm{d}t} = e \cdot MP_i - u_i \cdot M_eP_i \tag{A.74}$$

$$\frac{\mathrm{d}M_e}{\mathrm{d}t} = u_b \cdot M_eP_b + u_n \cdot M_eP_n + \sum_i u_i \cdot M_eP_i - d_{M_e}M_e \tag{A.75}$$

Note, the index i denotes the length of the exogenous peptide, so P_{i+1} is the peptide P_i with one amino acid extended from the N-terminus.

A.3.2 Model parameters

Table A.1. Parameters and values used for simulation of the mechanistic model.

Param	Description	Value Used	Units	Source
g_{self}	Supply rate of endogenous peptide	2×10^6	m.p.s.	[169]
g_i	Supply rate of exogenous peptide i	peptide specific	m.p.s.	♠
d_C	Degradation rate of peptide	1.572×10^{-3}	s^{-1}	♡
r_i	Cytosolic aminopeptidase trimming rate	peptide specific	s^{-1}	♡
u_S	Peptide-TAP off rate	4.681	s^{-1}	♡
k_S	Peptide-TAP turnover rate	5	s^{-1}	[109]
b^S_{self}	Endogenous peptide–TAP binding rate	1.159×10^{-8}	m.p.s.	
b^S_i	Peptide i TAP binding rate	peptide specific	m.p.s.	♢
E_0	Total ERAP1 molecules	1.536×10^7	-	♠
k_i	Peptide i turnover rate by ERAP1	peptide-specific	s^{-1}	♡
K_M	ERAP1 Michaelis term	100	μM	‡
g_M	MHC class I production rate	5.592	m.p.s.	♡
g_T	Tapasin production rate	1505	m.p.s.	[43]
ρ_n	Proportion of non-binding self peptide	9.994×10^{-1}	-	♡
ρ_b	Proportion of binding self peptide	5.860×10^{-4}	-	$1 - \rho_n$
d_M	MHC class I degradation rate	7.989×10^{-5}	s^{-1}	[43]
d_T	Tapasin degradation rate	1.726×10^{-3}	s^{-1}	[43]
d_P	Peptide degradation rate in ER	0.13	s^{-1}	[43]
b_T	MHC-Tapasin binding rate	1.099×10^{-9}	p.m.p.s.	[43]
b	Peptide-MHC binding rate	allele specific	p.m.p.s.	♣
c	Peptide-TM binding rate	9.630×10^{-6}	m.p.s.	♡
u_T	MHC-Tapasin off rate	1.185×10^{-6}	s^{-1}	[43]
u_i	Exogenous peptide i off rate in MHC-I	peptide-specific	s^{-1}	‡
u_n	Low affinity endogenous peptide off rate	1.433×10^{-3}	s^{-1}	♡
u_b	High affinity endogenous peptide off rate	5.844×10^{-6}	s^{-1}	♡
q	Tapasin enhancement to peptide-MHC off rate	21,035	-	[43]
v	Tapasin off rate increase factor with MP	936.3	-	[43]
e	Peptide-MHC egress rate	7.359×10^{-3}	s^{-1}	♣
d_{Me}	Empty surface MHC degradation rate	6.152×10^{-5}	s^{-1}	♡

Key: ♡ fitted to Hearn data; ♣ fitted to Dalchau data; ♠ proteasome model predictions; ♢ *PanTAP* model prediction; † ERAP1 model predictions; ‡ *NetMHC-4.0* predictions. Abbreviation: (p.)m.p.s = (per) molecules per second.

A.4 POEM supplementary figures

A.4.1 MHC-I representations

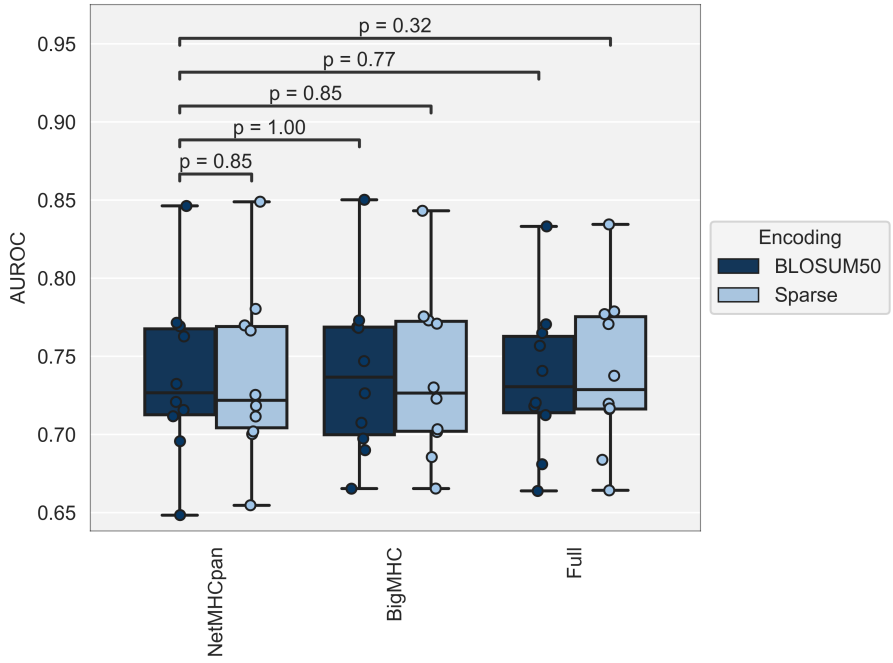


Figure A.1. MHC-I representations and associated amino acid encoding strategies compared by 10-fold cross-validation on *PRIME-2.0* dataset. Annotated p-values indicate the result of a two-tailed paired Wilcoxon signed-rank test between AUROC scores.

A.4.2 Peptide representations

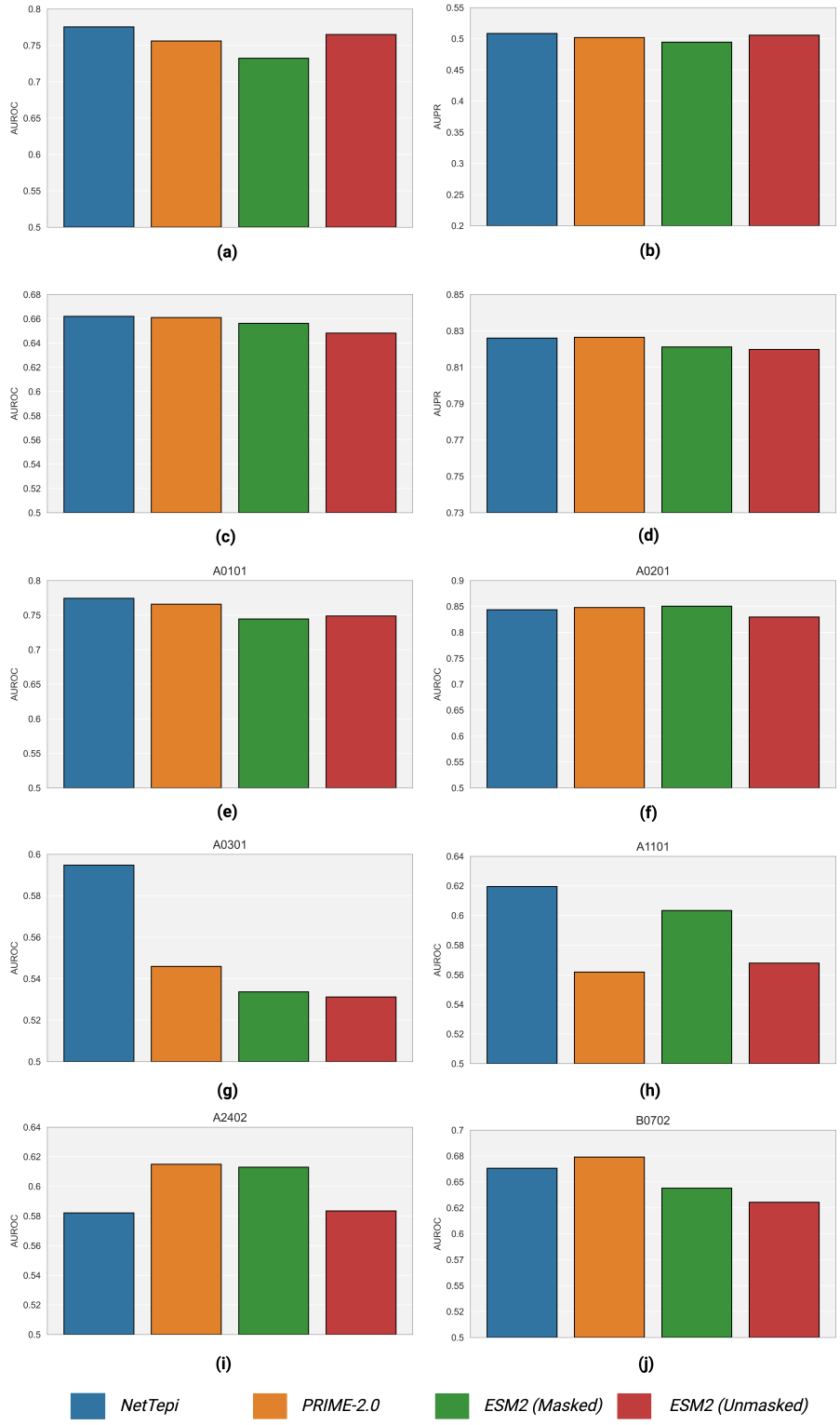


Figure A.2. Comparison of POEM performance when trained using 4 different encodings of the peptide sequence and tested on (a)-(b) GBM dataset, (c)-(d) IEDB pathogen dataset, and (e)-(j) SARS-CoV-2 dataset.

A.4.3 PRIME-2.0 dataset AUPR

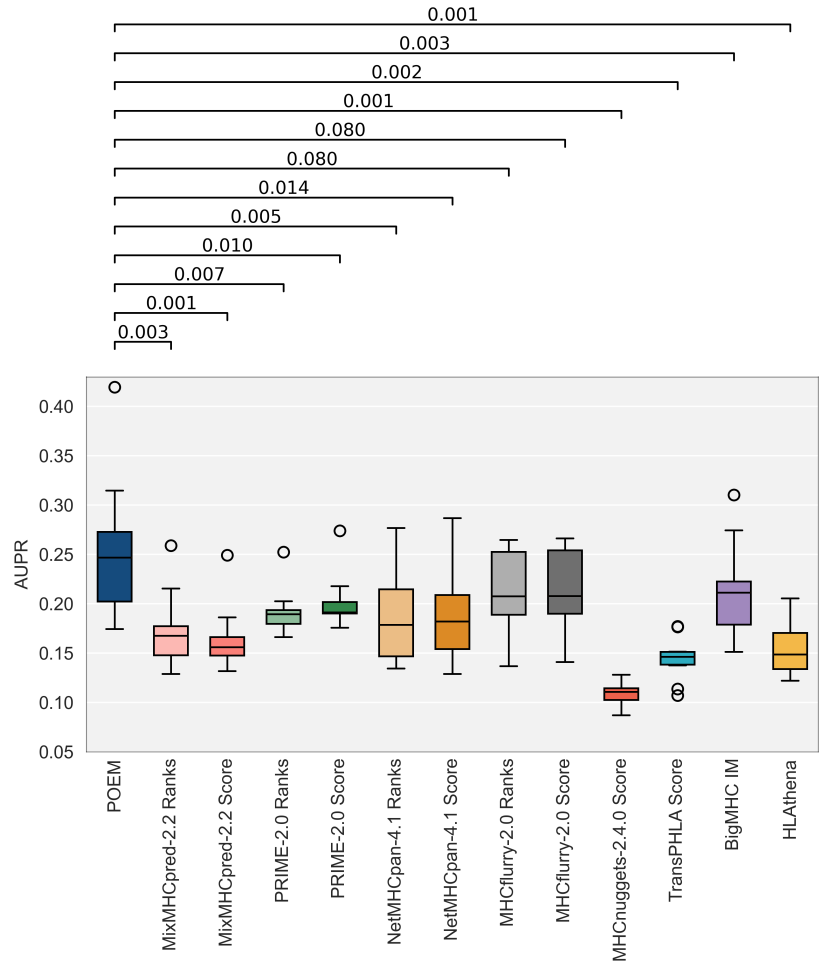


Figure A.3. Performance of POEM compared with the models in Table 7.1 via a 10-fold cross-validation and using the AUPR metric. A one-tailed paired Wilcoxon signed-rank test was used to test whether the AUPR of the other models was significantly lower than that of POEM. The resulting p-values are shown above the plot.

A.4.4 SARS-CoV-2 dataset AUPR

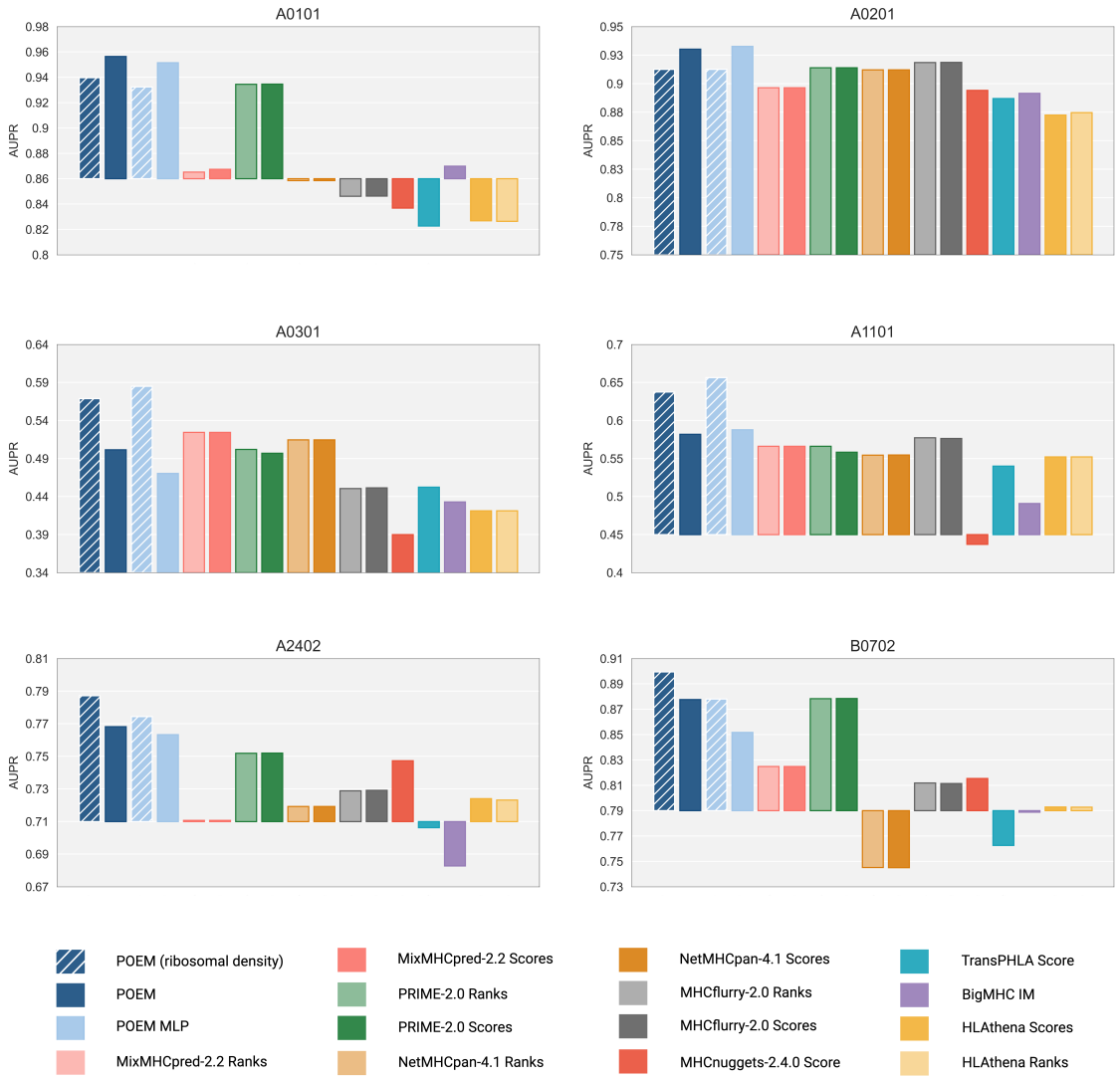


Figure A.4. POEM AUPR compared against other prominent immunogenicity predictors for a dataset of SARS-CoV-2 epitopes with experimentally validated immunogenicity status across the 6 most abundant alleles in IEDB.

A.4.5 POEM pMHC predictions comparison

Figure A.5. Comparison of POEM performance after re-training, replacing mechanistic model prediction with machine learning algorithm predictions of pMHC abundance.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. `https://www.tensorflow.org/`, 2015.

[2] Rupert Abele and Robert Tampé. The ABCs of Immunology: Structure and Function of TAP, the Transporter Associated with Antigen Processing. *Physiology*, 19(4):216–224, August 2004.

[3] Rupert Abele and Robert Tampé. Modulation of the antigen transport machinery TAP by friends and enemies. *FEBS Letters*, 580(4):1156–1163, February 2006.

[4] Esam T. Abualrous, Susanne Fritzsche, Zeynep Hein, Mohammed S. Al-Balushi, Peter Reinink, Louise H. Boyle, Ursula Wellbrock, Antony N. Antoniou, and Sebastian Springer. F pocket flexibility influences the tapasin dependence of two differentially disease-associated MHC Class I proteins. *European Journal of Immunology*, 45(4):1248–1257, April 2015.

[5] Yoshiki Akatsuka. TCR-Like CAR-T Cells Targeting MHC-Bound Minor Histocompatibility Antigens. *Frontiers in Immunology*, 11:257, February 2020.

[6] Nadja Akkad, Mark Schatz, Jörn Dengjel, Stefan Tenzer, and Hansjörg Schild. Census of cytosolic aminopeptidase activity reveals two novel cytosolic aminopeptidases. *Medical Microbiology and Immunology*, 201(4):463–473, November 2012.

[7] Maise Al Bakir, Ariana Huebner, Carlos Martínez-Ruiz, Kristiana Grigoriadis, Thomas B. K. Watkins, Oriol Pich, David A. Moore, Selvaraju Veeriah, Sophia Ward, Joanne Laycock, Diana Johnson, Andrew Rowan, Maryam Razaq, Mita Akther, Cristina Naceur-Lombardelli, Paulina Prymas, Antonia Toncheva, Sonya Hessey, Michelle Dietzen, Emma Colliver, Alexander M. Frankell, Abigail Bunkum, Emilia L. Lim, Takahiro Karasaki, Christopher Abbosh, Crispin T. Hiley, Mark S. Hill, Daniel E. Cook, Gareth A. Wilson, Roberto Salgado, Emma Nye, Richard Kevin Stone, Dean A. Fennell, Gillian Price, Keith M. Kerr, Babu Naidu, Gary Middleton, Yvonne Summers, Colin R. Lindsay, Fiona H. Blackhall, Judith Cave, Kevin G. Blyth, Arjun Nair, Asia Ahmed, Magali N. Taylor, Alexander James Procter, Mary Falzon, David Lawrence, Neal Navani, Ricky M. Thakrar, Sam M. Janes, Dionysis Papadatos-Pastos, Martin D. Forster, Siow Ming Lee, Tanya Ahmad, Sergio A. Quezada, Karl S. Peggs, Peter Van Loo, Caroline Dive, Allan Hackshaw, Nicolai J. Birkbak, Simone Zaccaria, TRACERx Consortium, Jason F. Lester, Amrita Bajaj, Apostolos Nakas, Azmina Sodha-Ramdeen, Keng Ang, Mohamad Tufail, Mohammed Fiyaz Chowdhry, Molly Scotland, Rebecca Boyles, Sridhar Rathinam, Claire Wilson, Domenic Marrone, Sean Dulloo, Gurdeep Matharu, Jacqui A. Shaw, Joan Riley, Lindsay Primrose, Ekaterini Boleti, Heather Cheyne, Mohammed Khalil, Shirley Richardson, Tracey Cruickshank, Sarah Benafif, Kayleigh Gilbert, Akshay J. Patel, Aya Osman, Christer Lacson, Gerald Langman, Helen Shackleford, Madava Djearaman, Salma Kadiri, Angela Leek, Jack Davies Hodgkinson, Nicola Totten, Angeles Montero, Elaine Smith, Eustace Fontaine, Felice Granato, Helen Doran, Juliette Novasio, Kendadai Rammohan, Leena Joseph, Paul Bishop,

Rajesh Shah, Stuart Moss, Vijay Joshi, Philip Crosbie, Fabio Gomes, Kate Brown, Mathew Carter, Anshuman Chaturvedi, Lynsey Priest, Pedro Oliveira, Matthew G. Krebs, Alexandra Clipson, Jonathan Tugwood, Alastair Kerr, Dominic G. Rothwell, Elaine Kilgour, Hugo J. W. L. Aerts, Roland F. Schwarz, Tom L. Kaufmann, Rachel Rosenthal, Zoltan Szallasi, Judit Kisistok, Mateo Sokac, Miklos Diossy, Jonas Demeulemeester, Aengus Stewart, Alastair Magness, Angeliki Karamani, Benny Chain, Brittany B. Campbell, Carla Castignani, Chris Bailey, Clare Puttick, Clare E. Weeden, Claudia Lee, Corentin Richard, David R. Pearce, Despoina Karagianni, Dhruva Biswas, Dina Levi, Elena Hoxha, Elizabeth Larose Cadieux, Eva Grönroos, Felip Gálvez-Cancino, Foteini Athanasopoulou, Francisco Gimeno-Valiente, George Kassiotis, Georgia Stavrou, Gerasimos Mastrokalos, Haoran Zhai, Helen L. Lowe, Ignacio Matos, Jacki Goldman, James L. Reading, James R. M. Black, Javier Herrero, Jayant K. Rane, Jerome Nicod, Jie Min Lam, John A. Hartley, Katey S. S. Enfield, Kayalvizhi Selvaraju, Kerstin Thol, Kevin Litchfield, Kevin W. Ng, Kezhong Chen, Krijn Dijkstra, Krupa Thakkar, Leah Ensell, Mansi Shah, Marcos Vasquez, Maria Litovchenko, Mariana Werner Sunderland, Michelle Leung, Mickael Escudero, Mihaela Angelova, Miljana Tanić, Monica Sivakumar, Nnennaya Kanu, Olga Chervova, Olivia Lucas, Othman Al-Sawaf, Philip Hobson, Piotr Pawlik, Robert Bentham, Robert E. Hynds, Roberto Vendramin, Sadegh Saghafinia, Saioa López, Samuel Gamble, Seng Kuong Anakin Ung, Sharon Vanloo, Stefan Boeing, Stephan Beck, Supreet Kaur Bola, Tamara Denner, Teresa Marafioti, Thanos P. Mourikis, Victoria Spanswick, Vittorio Barbè, Wei-Ting Lu, William Hill, Wing Kin Liu, Yin Wu, Yutaka Naito, Zoe Ramsden, Catarina Veiga, Gary Royle, Charles-Antoine Collins-Fekete, Francesco Fraioli, Paul Ashford, Tristan Clark, Elaine Borg, James Wilson, Davide Patrini, Emilie Martinoni Hoogenboom, Fleur Monk, James W. Holding, Junaid Choudhary, Kunal Bhakhri, Marco Scarci, Martin Hayward, Nikolaos Panagiotopoulos, Pat Gorman, Reena Khiroya, Robert C. M. Stephens, Yien Ning Sophia Wong, Steve Bandula, Abigail Sharp, Sean Smith, Nicole Gower, Harjot Kaur Dhanda, Kitty Chan, Camilla Pilotti, Rachel Leslie, Anca Grapa,

Hanyun Zhang, Khalid AbdulJabbar, Xiaoxi Pan, Yinyin Yuan, David Chuter, Mairead MacKenzie, Serena Chee, Aiman Alzetani, Lydia Scarlett, Jennifer Richards, Papawadee Ingram, Silvia Austin, Eric Lim, Paulo De Sousa, Simon Jordan, Alexandra Rice, Hilgardt Raubenheimer, Harshil Bhayani, Lyn Ambrose, Anand Devaraj, Hema Chavan, Sofina Begum, Silviu I. Buderi, Daniel Kaniu, Mpho Malima, Sarah Booth, Andrew G. Nicholson, Nadia Fernandes, Pratibha Shah, Chiara Proli, Madeleine Hewish, Sarah Danson, Michael J. Shackcloth, Lily Robinson, Peter Russell, Craig Dick, John Le Quesne, Alan Kirk, Mo Asif, Rocco Bilancia, Nikos Kostoulas, Mathew Thomas, Mariam Jamal-Hanjani, Nicholas McGranahan, and Charles Swanton. The evolution of non-small cell lung cancer metastases in TRACERx. *Nature*, 616(7957):534–542, April 2023.

[8] Anas Al-okaily, Razan Abu Khashabeh, Osama Alsmadi, Yazan Ahmad, Iyad Sultan, Ion Mandoiu, Abdelghani Tbakhi, and Pramod Srivast. ERAMER: A Novel In silico Tool for Prediction of ERAP1 Enzyme Trimming. preprint, In Review, January 2023.

[9] Saad Awadh Alanazi, M. M. Kamruzzaman, Md Nazirul Islam Sarker, Madallah Alruwaili, Yousef Alhwaiti, Nasser Alshammari, and Muhammad Hameed Siddiqi. Boosting Breast Cancer Detection Using Convolutional Neural Network. *Journal of Healthcare Engineering*, 2021:1–11, April 2021.

[10] Benjamin Alexander Albert, Yunxiao Yang, Xiaoshan M. Shao, Dipika Singh, Kellie N. Smith, Valsamo Anagnostou, and Rachel Karchin. Deep neural networks predict class I major histocompatibility complex epitope presentation and transfer learn neoepitope immunogenicity. *Nature Machine Intelligence*, 5(8):861–872, July 2023.

[11] Lindsey R. Baden, Hana M. El Sahly, Brandon Essink, Karen Kotloff, Sharon Frey, Rick Novak, David Diemert, Stephen A. Spector, Nadine Rouphael, C. Buddy Creech, John McGettigan, Shishir Khetan, Nathan Segall, Joel Solis, Adam Brosz, Carlos Fierro, Howard Schwartz, Kathleen Neuzil, Lawrence Corey, Peter Gilbert, Holly Janes, Dean Follmann, Mary Marovich, John Mas-

cola, Laura Polakowski, Julie Ledgerwood, Barney S. Graham, Hamilton Bennett, Rolando Pajon, Conor Knightly, Brett Leav, Weiping Deng, Honghong Zhou, Shu Han, Melanie Ivarsson, Jacqueline Miller, and Tal Zaks. Efficacy and Safety of the mRNA-1273 SARS-CoV-2 Vaccine. *New England Journal of Medicine*, 384(5):403–416, February 2021.

[12] Alistair Bailey, Neil Dalchau, Rachel Carter, Stephen Emmott, Andrew Phillips, Jörn M. Werner, and Tim Elliott. Selector function of MHC I molecules is determined by protein plasticity. *Scientific Reports*, 5:1–15, 2015. Publisher: Nature Publishing Group.

[13] Dominic J. Barker, Giuseppe Maccari, Xenia Georgiou, Michael A. Cooper, Paul Flicek, James Robinson, and Steven G. E. Marsh. The IPD-IMGT/HLA Database. *Nucleic Acids Research*, 51(D1):D1053–D1060, January 2023.

[14] Arman A. Bashirova, Mathias Viard, Vivek Naranbhai, Alba Grifoni, Wilfredo Garcia-Beltran, Marjan Akdag, Yuko Yuki, Xiaojiang Gao, Colm O'hUigin, Malini Raghavan, Steven Wolinsky, Jay H. Bream, Priya Duggal, Jeremy Martinson, Nelson L. Michael, Gregory D. Kirk, Susan P. Buchbinder, David Haas, James J. Goedert, Steven G. Deeks, Jacques Fellay, Bruce Walker, Philip Goulder, Peter Cresswell, Tim Elliott, Alessandro Sette, Jonathan Carlson, and Mary Carrington. HLA tapasin independence: broader peptide repertoire and HIV control. *Proceedings of the National Academy of Sciences of the United States of America*, 117(45):28232–28238, 2020. ISBN: 2013554117.

[15] Michal Bassani-Sternberg, Chloé Chong, Philippe Guillaume, Marthe Solleder, HuiSong Pak, Philippe O. Gannon, Lana E. Kandalaft, George Coukos, and David Gfeller. Deciphering HLA-I motifs across HLA peptidomes improves neoantigen predictions and identifies allostery regulating HLA specificity. *PLoS computational biology*, 13(8):e1005725, August 2017.

[16] Michal Bassani-Sternberg, Sune Pletscher-Frankild, Lars Juhl Jensen, and Matthias Mann. Mass Spectrometry of Human Leukocyte Antigen Class I Pep-

tidomes Reveals Strong Effects of Protein Abundance and Turnover on Antigen Presentation. *Molecular & Cellular Proteomics*, 14(3):658–673, March 2015.

[17] James Bergstra and Yoshua Bengio. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 2011.

[18] Dikla Berko, Shira Tabachnick-Cherny, Dalit Shental-Bechor, Paolo Cascio, Silvia Mioletti, Yaakov Levy, Arie Admon, Tamar Ziv, Boaz Tirosh, Alfred L. Goldberg, and Ami Navon. The Direction of Protein Entry into the Proteasome Determines the Variety of Products and Depends on the Force Needed to Unfold Its Two Termini. *Molecular Cell*, 48(4):601–611, November 2012.

[19] M. Bhasin and G. P. S. Raghava. Pcleavage: an SVM based method for prediction of constitutive proteasome and immunoproteasome cleavage sites in antigenic sequences. *Nucleic Acids Research*, 33(Web Server):W202–W207, July 2005.

[20] Manoj Bhasin and G.p.s. Raghava. Analysis and prediction of affinity of TAP binding peptides using cascade SVM. *Protein Science*, 13(3):596–607, 2004.

[21] Maxim Borisyak, Stefan Born, Peter Neubauer, and Mariano Nicolas Cruz-Bournazou. Deep Learning for Fast Inference of Mechanistic Models' Parameters. *arXiv*, 2023. Publisher: [object Object] Version Number: 1.

[22] Denise S.M. Boulanger, Ruth C. Eccleston, Andrew Phillips, Peter V. Coveney, Tim Elliott, and Neil Dalchau. A mechanistic model for predicting cell surface presentation of competing peptides by MHC class I molecules. *Frontiers in Immunology*, 9(JUL), 2018.

[23] Nathalie Brouwenstijn, Thomas Serwold, and Nilabh Shastri. MHC Class I Molecules Can Direct Proteolytic Cleavage of Antigenic Precursors in the Endoplasmic Reticulum. *Immunity*, 15(1):95–104, July 2001.

[24] Liam V. Brown, Eamonn A. Gaffney, Jonathan Wagg, and Mark C. Coles. An in silico model of cytotoxic T-lymphocyte activation in the lymph node

following short peptide vaccination. *Journal of the Royal Society Interface*, 15(140), 2018. Publisher: Royal Society Publishing.

[25] Liam V Brown, Jonathan Wagg, Rachel Darley, Andy Van Hateren, Tim Elliott, Eamonn A Gaffney, and Mark C Coles. De-risking clinical trial failure through mechanistic simulation. *Immunotherapy Advances*, 2(1):ltac017, January 2022.

[26] R. Bruno, D. Hille, A. Riva, N. Vivier, W. W. ten Bokkel Huinnink, A. T. van Oosterom, S. B. Kaye, J. Verweij, F. V. Fossella, V. Valero, J. R. Rigas, A. D. Seidman, B. Chevallier, P. Fumoleau, H. A. Burris, P. M. Ravdin, and L. B. Sheiner. Population pharmacokinetics/pharmacodynamics of docetaxel in phase II studies in patients with cancer. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology*, 16(1):187–196, January 1998.

[27] V. Brusic, P. van Endert, J. Zeleznikow, S. Daniel, J. Hammer, and N. Petrovsky. A neural network model approach to the study of human TAP transporter. *In Silico Biology*, 1(2):109–121, 1999.

[28] Patrick Bryant, Gabriele Pozzati, and Arne Elofsson. Improved prediction of protein-protein interactions using AlphaFold2. *Nature Communications*, 13(1):1265, March 2022.

[29] Paul R Buckley, Chloe H Lee, Ruichong Ma, Isaac Woodhouse, Jeongmin Woo, Vasily O Tsvetkov, Dmitrii S Shcherbinin, Agne Antanaviciute, Mikhail Shughay, Margarida Rei, Alison Simmons, and Hashem Koohy. Evaluating performance of existing computational models in predicting CD8+ T cell pathogenic epitopes and cancer neoantigens. *Briefings in Bioinformatics*, 23(3):bbac141, May 2022.

[30] Brendan Bulik-Sullivan, Jennifer Busby, Christine D Palmer, Matthew J Davis, Tyler Murphy, Andrew Clark, Michele Busby, Fujiko Duke, Aaron Yang, Lauren Young, Noelle C Ojo, Kamilah Caldwell, Jesse Abhyankar, Thomas Boucher, Meghan G Hart, Vladimir Makarov, Vincent Thomas De Montpreville, Olaf

Mercier, Timothy A Chan, Giorgio Scagliotti, Paolo Bironzo, Silvia Novello, Niki Karachaliou, Rafael Rosell, Ian Anderson, Nashat Gabrail, John Hrom, Chainarong Limvarapuss, Karin Choquette, Alexander Spira, Raphael Rousseau, Cynthia Voong, Naiyer A Rizvi, Elie Fadel, Mark Frattini, Karin Jooss, Mojca Skoberne, Joshua Francis, and Roman Yelensky. Deep learning using tumor HLA peptide mass spectrometry datasets improves neoantigen identification. *Nature Biotechnology*, 37(1):55–63, January 2019.

[31] Anne Burgevin, Loredana Saveanu, Yohan Kim, Émilie Barilleau, Maya Kotturi, Alessandro Sette, Peter van Endert, and Bjoern Peters. A detailed analysis of the murine TAP transporter substrate specificity. *PLoS ONE*, 3(6):1–8, 2008.

[32] Jorg J. A. Calis, Matt Maybeno, Jason A. Greenbaum, Daniela Weiskopf, Aruna D. De Silva, Alessandro Sette, Can Keşmir, and Bjoern Peters. Properties of MHC Class I Presented Peptides That Enhance Immunogenicity. *PLoS Computational Biology*, 9(10):e1003266, October 2013.

[33] Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. *Stan* : A Probabilistic Programming Language. *Journal of Statistical Software*, 76(1), 2017.

[34] Paolo Cascio, Craig Hilton, Alexei F. Kisselev, Kenneth L. Rock, and Alfred L. Goldberg. 26S proteasomes and immunoproteasomes produce mainly N-extended versions of an antigenic peptide. *EMBO Journal*, 20(10):2357–2366, 2001.

[35] Shih-Chung Chang, Frank Momburg, Nidhi Bhutani, and Alfred L. Goldberg. The ER aminopeptidase, ERAP1, trims precursors to lengths of MHC class I peptides by a "molecular ruler" mechanism. *Proceedings of the National Academy of Sciences*, 102(47):17107–17112, November 2005.

[36] Spencer A. Chang, Vashti G. Lacaille, David S. Guttoh, and Matthew J. Androlewicz. Binding and transport of melanoma-specific antigenic peptides by

the transporter associated with antigen processing. *Molecular Immunology*, 33(15):1165–1169, October 1996.

[37] Brad Chapman and Jeffrey Chang. Biopython: Python tools for computational biology. *ACM SIGBIO Newsletter*, 20(2):15–19, August 2000.

[38] Boseung Choi, Grzegorz A. Rempala, and Jae Kyoung Kim. Beyond the Michaelis-Menten equation: Accurate and efficient estimation of enzyme kinetic parameters. *Scientific Reports*, 7(1):1–11, 2017. Publisher: Springer US.

[39] Yanyi Chu, Yan Zhang, Qiankun Wang, Lingfeng Zhang, Xuhong Wang, Yanjing Wang, Dennis Russell Salahub, Qin Xu, Jianmin Wang, Xue Jiang, Yi Xiong, and Dong-Qing Wei. A transformer-based model to predict peptide–HLA class I binding and optimize mutated peptides for vaccine design. *Nature Machine Intelligence*, 4(3):300–311, March 2022.

[40] Michael Clerx, Martin Robinson, Ben Lambert, Chon Lok Lei, Sanmitra Ghosh, Gary R. Mirams, and David J. Gavaghan. Probabilistic Inference on Noisy Time Series (PINTS). *Journal of Open Research Software*, 7(1):23, July 2019.

[41] L. S. Cohen and G. P. Studzinski. Correlation between cell enlargement and nucleic acid and protein content of HeLa cells in unbalanced growth produced by inhibitors of DNA synthesis. *Journal of Cellular Physiology*, 69(3):331–339, June 1967.

[42] Geoffrey M. Cooper. *The cell: a molecular approach*. ASM Press [u.a.], Washington, DC, 2. ed edition, 2000.

[43] Neil Dalchau, Andrew Phillips, Leonard D. Goldstein, Mark Howarth, Luca Cardelli, Stephen Emmott, Tim Elliott, and Joern M. Werner. A peptide filtering relation quantifies MHC class I peptide optimization. *PLoS Computational Biology*, 7(10), 2011.

[44] Soizic Daniel, Vladimir Brusic, Sophie Caillat-Zucman, Nicolai Petrovsky, Leonard Harrison, Daniela Riganelli, Francesco Sinigaglia, Fabio Gallazzi,

Jürgen Hammer, and Peter M. Van Endert. Relationship Between Peptide Selectivities of Human Transporters Associated with Antigen Processing and HLA Class I Molecules. *The Journal of Immunology*, 161(2):617–624, July 1998.

[45] José A.López de Castro and Efstratios Stratikos. Intracellular antigen processing by ERAP2: Molecular mechanism and roles in health and disease. *Human Immunology*, 80(5):310–317, 2019. Publisher: Elsevier.

[46] Janosch Deeg, Markus Axmann, Jovana Matic, Anastasia Liapis, David Depoil, Jehan Afrose, Silvia Curado, Michael L. Dustin, and Joachim P. Spatz. T Cell Activation is Determined by the Number of Presented Antigens. *Nano Letters*, 13(11):5619–5626, November 2013.

[47] Hendrik Dietz and Matthias Rief. Protein structure by mechanical triangulation. *Proceedings of the National Academy of Sciences*, 103(5):1244–1247, January 2006.

[48] Carmen M. Diez-Rivero, Bernardo Chenlo, Pilar Zuluaga, and Pedro A. Reche. Quantitative modeling of peptide binding to TAP using support vector machine. *Proteins: Structure, Function and Bioinformatics*, 78(1):63–72, 2010.

[49] Emilio Dorigatti, Bernd Bischl, and Benjamin Schubert. Improved proteasomal cleavage prediction with positive-unlabeled learning, October 2022. arXiv:2209.07527 [cs, q-bio].

[50] Irini Doytchinova, Shelley Hemsley, and Darren R. Flower. Transporter Associated with Antigen Processing Preselection of Peptides Binding to the MHC: A Bioinformatic Evaluation. *The Journal of Immunology*, 173(11):6813–6819, December 2004.

[51] Freeman Dyson. A meeting with Enrico Fermi. *Nature*, 427(6972):297–297, January 2004.

[52] Pierre Dönnes and Oliver Kohlbacher. Integrated modeling of the major events

in the MHC class I antigen processing pathway. *Protein Science*, 14(8):2132–2140, 2005.

[53] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, October 2022.

[54] Irini Evnouchidou, Ram Kamal, Sergey Seregin, Yoshikuni Goto, Masafumi Tsujimoto, Akira Hattori, Paraskevi Voulgari, Alexandros Drosos, Andrea Amalfitano, Ian York, and Efstratios Stratikos. Cutting Edge: Coding Single Nucleotide Polymorphisms of Endoplasmic Reticulum Aminopeptidase 1 Can Affect Antigenic Peptide Generation In Vitro by Influencing Basic Enzymatic Properties of the Enzyme. *Journal of immunology (Baltimore, Md. : 1950)*, 186:1909–13, February 2011.

[55] Irini Evnouchidou, Frank Momburg, Athanasios Papakyriakou, Angeliki Chroni, Leondios Leondiadis, Shih-Chung Chang, Alfred L. Goldberg, and Efstratios Stratikos. The Internal Sequence of the Peptide-Substrate Determines Its N-Terminus Trimming by ERAP1. *PLoS ONE*, 3(11):e3658, November 2008.

[56] K. Falk, O. Rötzschke, and H. G. Rammensee. Cellular peptide composition governed by major histocompatibility complex class I molecules. *Nature*, 348(6298):248–251, November 1990.

[57] Deborah A. Ferrington and Dale S. Gregerson. Immunoproteasomes: structure, function, and antigen presentation. *Progress in Molecular Biology and Translational Science*, 109:75–112, 2012.

[58] Yaara Finkel, Orel Mizrahi, Aharon Nachshon, Shira Weingarten-Gabbay, David Morgenstern, Yfat Yahalom-Ronen, Hadas Tamir, Hagit Achdout, Dana Stein, Ofir Israeli, Adi Beth-Din, Sharon Melamed, Shay Weiss, Tomer Israely, Nir Paran, Michal Schwartz, and Noam Stern-Ginossar. The coding capacity of

SARS-CoV-2. *Nature*, 589(7840):125–130, January 2021. Number: 7840 Publisher: Nature Publishing Group.

[59] Katharina Fleischhauer, Doriana Fruci, Peter van Endert, Jean Herman, Silvia Tanzarella, Hans-J. Wallny, Pierre Coulie, Claudio Bordignon, and Catia Traversari. Characterization of antigenic peptides presented by HLA-B44 molecules on tumor cells expressing the gene MAGE-3. *International Journal of Cancer*, 68(5):622–628, 1996.

[60] Angela Frentzen, Jason A. Greenbaum, Haeuk Kim, Bjoern Peters, and Zeynep Koşaloğlu-Yalçın. Estimating tissue-specific peptide abundance from public RNA-Seq data. *Frontiers in Genetics*, 14:1082168, 2023.

[61] Aki Fujioka, Kenta Terai, Reina E. Itoh, Kazuhiro Aoki, Takeshi Nakamura, Shinya Kuroda, Eisuke Nishida, and Michiyuki Matsuda. Dynamics of the Ras/ERK MAPK cascade as monitored by fluorescent probes. *The Journal of Biological Chemistry*, 281(13):8917–8926, March 2006.

[62] Amit Gandhi, Damodharan Lakshminarasimhan, Yixin Sun, and Hwai-Chen Guo. Structural insights into the molecular ruler mechanism of the endoplasmic reticulum aminopeptidase ERAP1. *Scientific Reports*, 1(1):186, December 2011.

[63] Heli M. Garcia Alvarez, Zeynep Koşaloğlu-Yalçın, Bjoern Peters, and Morten Nielsen. The role of antigen expression in shaping the repertoire of HLA presented ligands. *iScience*, 25(9):104975, September 2022.

[64] Malgorzata Anna Garstka, Susanne Fritzsche, Izabela Lenart, Zeynep Hein, Gytis Jankevicius, Louise H. Boyle, Tim Elliott, John Trowsdale, Antony N. Antoniou, Martin Zacharias, and Sebastian Springer. Tapasin dependence of major histocompatibility complex class I molecules correlates with their conformational flexibility. *The FASEB Journal*, 25(11):3989–3998, November 2011.

[65] Robert A. Gatenby and Edward T. Gawlinski. A reaction-diffusion model of cancer invasion. *Cancer Research*, 56(24):5745–5753, 1996.

[66] David Gfeller, Julien Schmidt, Giancarlo Croce, Philippe Guillaume, Sara Bobisse, Raphael Genolet, Lise Queiroz, Julien Cesbron, Julien Racle, and Alexandre Harari. Improved predictions of antigen presentation and TCR recognition with MixMHCpred2.2 and PRIME2.0 reveal potent SARS-CoV-2 CD8+ T-cell epitopes. *Cell Systems*, 14(1):72–83.e5, January 2023.

[67] Umesh Ghoshdastider and Ataman Sendoel. Exploring the pan-cancer landscape of posttranscriptional regulation. *Cell Reports*, 42(10):113172, October 2023.

[68] P. A. Gorer. The genetic and antigenic basis of tumour transplantation. *The Journal of Pathology and Bacteriology*, 44(3):691–697, May 1937.

[69] Brigitte Gubler, Soizic Daniel, Elena A Armandola, Juergen Hammer, Sophie Caillat-Zucman, Peter M.van Endert, and Inserm U. Substrate selection by transporters associated with antigen processing occurs during peptide binding to TAP. *Molecular Immunology*, 35(8):427–433, May 1998.

[70] Jean Hausser, Avi Mayo, Leeat Keren, and Uri Alon. Central dogma rates and the trade-off between precision and economy in gene expression. *Nature Communications*, 10(1):68, January 2019.

[71] Arron Hearn, Ian A. York, Courtney Bishop, and Kenneth L. Rock. Characterizing the Specificity and Cooperation of Aminopeptidases in the Cytosol and Endoplasmic Reticulum during MHC Class I Antigen Presentation. *The Journal of Immunology*, 184(9):4725–4732, 2010.

[72] Arron Hearn, Ian A. York, and Kenneth L. Rock. The Specificity of Trimming of MHC Class I-Presented Peptides in the Endoplasmic Reticulum. *The Journal of Immunology*, 183(9):5526–5536, 2009.

[73] M. T. Heemels and H. L. Ploegh. Substrate specificity of allelic variants of the TAP peptide transporter. *Immunity*, 1(9):775–784, December 1994.

[74] Meike Herget, Christoph Baldauf, Christian Schölz, David Parcej, Karl-Heinz Wiesmüller, Robert Tampé, Rupert Abele, and Enrica Bordignon. Conforma-

tion of peptides bound to the transporter associated with antigen processing (TAP). *Proceedings of the National Academy of Sciences*, 108(4):1349–1354, January 2011.

[75] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544, August 1952.

[76] Mark Howarth, Anthony Williams, Anne B. Tolstrup, and Tim Elliott. Tapasin enhances MHC class I peptide presentation according to peptide half-life. *Proceedings of the National Academy of Sciences of the United States of America*, 101(32):11737–11742, 2004.

[77] Eva M. Huber, Michael Basler, Ricarda Schwab, Wolfgang Heinemeyer, Christopher J. Kirk, Marcus Groettrup, and Michael Groll. Immuno- and Constitutive Proteasome Crystal Structures Reveal Differences in Substrate and Inhibitor Specificity. *Cell*, 148(4):727–738, February 2012.

[78] Jonathan P. Hutchinson, Ioannis Temponeras, Jonas Kuiper, Adrian Cortes, Justyna Korczynska, Semra Kitchen, and Efstratios Stratikos. Common allotypes of ER aminopeptidase 1 have substrate-dependent and highly variable enzymatic properties. preprint, Biochemistry, November 2020.

[79] Vanessa Jurtz, Sinu Paul, Massimo Andreatta, Paolo Marcatili, Bjoern Peters, and Morten Nielsen. NetMHCpan-4.0: Improved Peptide-MHC Class I Interaction Predictions Integrating Eluted Ligand and Peptide Binding Affinity Data. *Journal of Immunology (Baltimore, Md.: 1950)*, 199(9):3360–3368, November 2017.

[80] Kasper W. Jørgensen, Michael Rasmussen, Søren Buus, and Morten Nielsen. NetMHCstab – predicting stability of peptide–MHC-I complexes; impacts for cytotoxic T lymphocyte epitope discovery. *Immunology*, 141(1):18–26, 2014. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/imm.12160.

[81] S. Kawashima. AAindex: Amino Acid index database. *Nucleic Acids Research*, 28(1):374–374, January 2000.

[82] Can Kesmir, Vera Van Noort, Rob J. De Boer, and Paulien Hogeweg. Bioinformatic analysis of functional differences between the immunoproteasome and the constitutive proteasome. *Immunogenetics*, 55(7):437–449, October 2003.

[83] Can Keşmir, Alexander K. Nussbaum, Hansjörg Schild, Vincent Detours, and Søren Brunak. Prediction of proteasome cleavage motifs by neural networks. *Protein Engineering, Design and Selection*, 15(4):287–296, April 2002.

[84] Alexei F. Kisselev, Tatos N. Akopian, Kee Min Woo, and Alfred L. Goldberg. The Sizes of Peptides Generated from Protein by Mammalian 26 and 20 S Proteasomes. *Journal of Biological Chemistry*, 274(6):3363–3371, February 1999.

[85] Jennifer Klunk, Tauras P. Vilgalys, Christian E. Demeure, Xiaoheng Cheng, Mari Shiratori, Julien Madej, Rémi Beau, Derek Elli, Maria I. Patino, Rebecca Redfern, Sharon N. DeWitte, Julia A. Gamble, Jesper L. Boldsen, Ann Carmichael, Nükhet Varlik, Katherine Eaton, Jean-Christophe Grenier, G. Brian Golding, Alison Devault, Jean-Marie Rouillard, Vania Yotova, Renata Sindeaux, Chun Jimmie Ye, Matin Bikaran, Anne Dumaine, Jessica F. Brinkworth, Dominique Missiakas, Guy A. Rouleau, Matthias Steinrücken, Javier Pizarro-Cerdá, Hendrik N. Poinar, and Luis B. Barreiro. Evolution of immune genes is associated with the Black Death. *Nature*, 611(7935):312–319, November 2022.

[86] Despoina Koumantou, Eilon Barnea, Adrian Martin-Esteban, Zachary Maben, Athanasios Papakyriakou, Anastasia Mpakali, Paraskevi Kokkala, Harris Pratsinis, Dimitris Georgiadis, Lawrence J. Stern, Arie Admon, and Efstratios Stratikos. Editing the immunopeptidome of melanoma cells using a potent inhibitor of endoplasmic reticulum aminopeptidase 1 (ERAP1). *Cancer Immunology, Immunotherapy*, 68(8):1245–1261, August 2019.

[87] Zeynep Koşaloğlu-Yalçın, Jenny Lee, Jason Greenbaum, Stephen P. Schoenberger, Aaron Miller, Young J. Kim, Alessandro Sette, Morten Nielsen, and

Bjoern Peters. Combined assessment of MHC binding and antigen abundance improves T cell epitope predictions. *iScience*, 25(2):103850, February 2022.

[88] Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. *Machine Learning*, 51(2):181–207, 2003.

[89] C. Kuttler, A. K. Nussbaum, T. P. Dick, H. G. Rammensee, H. Schild, and K. P. Hadeler. An algorithm for the prediction of proteasomal cleavages. *Journal of Molecular Biology*, 298(3):417–429, May 2000.

[90] Alexandr Kuznetsov, Alice Voronina, Vadim Govorun, and Georgij Arapidi. Critical Review of Existing MHC I Immunopeptidome Isolation Methods. *Molecules*, 25(22):5409, November 2020.

[91] Thomas J. Lane. Protein structure prediction has reached the single-structure frontier. *Nature Methods*, 20(2):170–173, February 2023.

[92] Franziska Lang, Barbara Schrörs, Martin Löwer, Özlem Türeci, and Ugur Sahin. Identification of neoantigens for individualized therapeutic cancer vaccines. *Nature Reviews. Drug Discovery*, 21(4):261–282, April 2022.

[93] Mette V Larsen, Claus Lundegaard, Kasper Lamberth, Soren Buus, Soren Brunak, Ole Lund, and Morten Nielsen. An integrative approach to CTL epitope prediction: A combined algorithm integrating MHC class I binding, TAP transport efficiency, and proteasomal cleavage predictions. *European Journal of Immunology*, 35(8):2295–2303, August 2005.

[94] Mette V Larsen, Claus Lundegaard, Kasper Lamberth, Soren Buus, Ole Lund, and Morten Nielsen. Large-scale validation of methods for cytotoxic T-lymphocyte epitope prediction. *BMC Bioinformatics*, 8(1):424, December 2007.

[95] Sneh Lata, Manoj Bhasin, and Gajendra PS Raghava. MHCBN 4.0: A database of MHC/TAP binding peptides and T-cell epitopes. *BMC Research Notes*, 2(1):61, April 2009.

[96] Estibaliz Lazaro, Carl Kadie, Pamela Stamegna, Shao Chong Zhang, Pauline Gourdain, Nicole Y. Lai, Mei Zhang, Sergio A. Martinez, David Heckerman, and Sylvie Le Gall. Variable HIV peptide stability in human cytosol is critical to epitope presentation and immune escape. *The Journal of Clinical Investigation*, 121(6):2480–2492, June 2011.

[97] Chloe H. Lee, Jaesung Huh, Paul R. Buckley, Myeongjun Jang, Mariana Pereira Pinho, Ricardo A. Fernandes, Agne Antanaviciute, Alison Simmons, and Hashem Koohy. A robust deep learning platform to predict CD8+ T-cell epitopes. preprint, Bioinformatics, December 2022.

[98] Chloe H. Lee, Jaesung Huh, Paul R. Buckley, Myeongjun Jang, Mariana Pereira Pinho, Ricardo A. Fernandes, Agne Antanaviciute, Alison Simmons, and Hashem Koohy. A robust deep learning workflow to predict CD8+T-cell epitopes. *Genome Medicine*, 15(1):70, September 2023.

[99] James Lee, Michael L. Oldham, Victor Manon, and Jue Chen. Principles of peptide selection by the transporter associated with antigen processing. *Proceedings of the National Academy of Sciences*, 121(23):e2320879121, June 2024.

[100] Elisa Lehnert and Robert Tampé. Structure and Dynamics of Antigenic Peptides in Complex with TAP. *Frontiers in Immunology*, 8, January 2017.

[101] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan Dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023.

[102] B. Liu, Y. Dai, X. Li, W.S. Lee, and P.S. Yu. Building text classifiers using positive and unlabeled examples. In *Third IEEE International Conference on Data Mining*, pages 179–186, Melbourne, FL, USA, 2003. IEEE Comput. Soc.

[103] Maria Lucchiari-Hartz, Peter M. Van Endert, Grégoire Lauvau, Reinhard Maier, Andreas Meyerhans, Derek Mann, Klaus Eichmann, and Gabriele Niedermann.

Cytotoxic T Lymphocyte Epitopes of HIV-1 Nef. *The Journal of Experimental Medicine*, 191(2):239–252, January 2000.

[104] Claus Lundegaard, Ole Lund, Can Keşmir, Søren Brunak, and Morten Nielsen. Modeling the adaptive immune system: predictions and simulations. *Bioinformatics*, 23(24):3265–3275, December 2007.

[105] José A. López de Castro. How ERAP1 and ERAP2 Shape the Peptidomes of Disease-Associated MHC-I Proteins. *Frontiers in immunology*, 9(October):2463, 2018.

[106] Ruichong Ma, Margarida Rei, Isaac Woodhouse, Katherine Ferris, Sophie Kirschner, Anandhakumar Chandran, Uzi Gileadi, Ji-Li Chen, Mariana Pereira Pinho, Yoanna Ariosa-Morejon, Skirmantas Kriaucionis, Nicola Ternette, Hashem Koohy, Olaf Ansorge, Graham Ogg, Puneet Plaha, and Vincenzo Cerundolo. Decitabine increases neoantigen and cancer testis antigen expression to enhance T-cell–mediated toxicity against glioblastoma. *Neuro-Oncology*, 24(12):2093–2106, December 2022.

[107] Whitney A. Macdonald, Zhenjun Chen, Stephanie Gras, Julia K. Archbold, Fleur E. Tynan, Craig S. Clements, Mandvi Bharadwaj, Lars Kjer-Nielsen, Philippa M. Saunders, Matthew C. J. Wilce, Fran Crawford, Brian Stadinsky, David Jackson, Andrew G. Brooks, Anthony W. Purcell, John W. Kappler, Scott R. Burrows, Jamie Rossjohn, and James McCluskey. T cell allorecognition via molecular mimicry. *Immunity*, 31(6):897–908, December 2009.

[108] Simeone Marino, Ian B. Hogue, Christian J. Ray, and Denise E. Kirschner. A methodology for performing global uncertainty and sensitivity analysis in systems biology. *Journal of Theoretical Biology*, 254(1):178–196, September 2008.

[109] Pär Matsson, Luca A. Fenu, Patrik Lundquist, Jacek R. Wiśniewski, Manfred Kansy, and Per Artursson. Quantifying the impact of transporters on cellular drug permeability. *Trends in Pharmacological Sciences*, 36(5):255–262, May 2015.

[110] George Mavridis, Richa Arya, Alexander Domnick, Jerome Zoidakis, Manousos Makridakis, Antonia Vlahou, Anastasia Mpakali, Angelos Lelis, Dimitris Georgiadis, Robert Tampé, Athanasios Papakyriakou, Lawrence J. Stern, and Efstratios Stratikos. A systematic re-examination of processing of MHCI-bound antigenic peptide precursors by endoplasmic reticulum aminopeptidase 1. *Journal of Biological Chemistry*, 295(21):7193–7210, May 2020.

[111] George Mavridis, Anastasia Mpakali, Jerome Zoidakis, Manousos Makridakis, Antonia Vlahou, Eleni Kaloumenou, Angeliki Ziotopoulou, Dimitris Georgiadis, Athanasios Papakyriakou, and Efstratios Stratikos. The ERAP1 active site cannot productively access the N-terminus of antigenic peptide precursors stably bound onto MHC class I. *Scientific Reports*, 11(1):1–10, 2021. Publisher: Nature Publishing Group UK ISBN: 0123456789.

[112] Anastasia Mpakali, Petros Giastas, Nikolas Mathioudakis, Irene M. Mavridis, Emmanuel Saridakis, and Efstratios Stratikos. Structural Basis for Antigenic Peptide Recognition and Processing by Endoplasmic Reticulum (ER) Aminopeptidase 2. *Journal of Biological Chemistry*, 290(43):26021–26032, October 2015.

[113] Andrea T. Nguyen, Christopher Szeto, and Stephanie Gras. The pockets guide to HLA class I molecules. *Biochemical Society Transactions*, 49(5):2319–2331, November 2021.

[114] Morten Nielsen, Claus Lundegaard, Thomas Blicher, Kasper Lamberth, Mikkel Harndahl, Sune Justesen, Gustav Røder, Bjoern Peters, Alessandro Sette, Ole Lund, and Søren Buus. NetMHCpan, a Method for Quantitative Predictions of Peptide Binding to Any HLA-A and -B Locus Protein of Known Sequence. *PLoS ONE*, 2(8):e796, August 2007.

[115] Morten Nielsen, Claus Lundegaard, Ole Lund, and Can Keşmir. The role of the proteasome in generating cytotoxic T-cell epitopes: insights obtained from improved predictions of proteasomal cleavage. *Immunogenetics*, 57(1):33–41, April 2005.

[116] A. K. Nussbaum, T. P. Dick, W. Keilholz, M. Schirle, S. Stevanović, K. Dietz, W. Heinemeyer, M. Groll, D. H. Wolf, R. Huber, H. G. Rammensee, and H. Schild. Cleavage motifs of the yeast 20S proteasome beta subunits deduced from digests of enolase 1. *Proceedings of the National Academy of Sciences of the United States of America*, 95(21):12504–12509, October 1998.

[117] Timothy J. O'Donnell, Alex Rubinsteyn, Maria Bonsack, Angelika B. Riemer, Uri Laserson, and Jeff Hammerbacher. MHCflurry: Open-Source Class I MHC Binding Affinity Prediction. *Cell Systems*, 7(1):129–132.e4, July 2018.

[118] Timothy J. O'Donnell, Alex Rubinsteyn, and Uri Laserson. 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. Publisher: Elsevier Inc.

[119] L. E. M. Oosten, D. Koppers-Lalic, E. Blokland, A. Mulder, M. E. Ressing, T. Mutis, A. G. S. Van Halteren, E. J. H. J. Wiertz, and E. Goulmy. TAP-inhibiting proteins US6, ICP47 and UL49.5 differentially affect minor and major histocompatibility antigen-specific recognition by cytotoxic T lymphocytes. *International Immunology*, 19(9):1115–1122, September 2007.

[120] B. Ortmann, M. J. Androlewicz, and P. Cresswell. MHC class I/beta 2-microglobulin complexes associate with TAP transporters before peptide binding. *Nature*, 368(6474):864–867, April 1994.

[121] Patrick A. Ott, Zhuting Hu, Derin B. Keskin, Sachet A. Shukla, Jing Sun, David J. Bozym, Wandi Zhang, Adrienne Luoma, Anita Giobbie-Hurder, Lauren Peter, Christina Chen, Oriol Olive, Todd A. Carter, Shuqiang Li, David J. Lieb, Thomas Eisenhaure, Evisa Gjini, Jonathan Stevens, William J. Lane, Indu Javeri, Kaliappanadar Nellaiappan, Andres M. Salazar, Heather Daley, Michael Seaman, Elizabeth I. Buchbinder, Charles H. Yoon, Maegan Harden, Niall Lennon, Stacey Gabriel, Scott J. Rodig, Dan H. Barouch, Jon C. Aster, Gad Getz, Kai Wucherpfennig, Donna Neuberg, Jerome Ritz, Eric S. Lander, Edward F. Fritsch, Nir Hacohen, and Catherine J. Wu. An immunogenic per-

sonal neoantigen vaccine for patients with melanoma. *Nature*, 547(7662):217–221, July 2017.

[122] Athanasios Papakyriakou, Emma Reeves, Mary Beton, Halina Mikolajek, Leon Douglas, Grace Cooper, Tim Elliott, Jörn M. Werner, and Edward James. The partial dissociation of MHC class I-bound peptides exposes their N terminus to trimming by endoplasmic reticulum aminopeptidase 1. *The Journal of Biological Chemistry*, 293(20):7538–7548, May 2018.

[123] Tracey Papenfuss and Brad Bolon. Lymphocytes. In Hans-Werner Vohr, editor, *Encyclopedia of Immunotoxicology*, pages 1–8. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[124] K. C. Parker, M. A. Bednarek, and J. E. Coligan. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *Journal of Immunology (Baltimore, Md.: 1950)*, 152(1):163–175, January 1994.

[125] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, and David Cournapeau. Scikit-learn: Machine Learning in Python. *MACHINE LEARNING IN PYTHON*, October 2011.

[126] Björn Peters, Sascha Bulik, Robert Tampe, Peter M. van Endert, and Hermann-Georg Holzhütter. Identifying MHC Class I Epitopes by Predicting the TAP Transport Efficiency of Epitope Precursors. *The Journal of Immunology*, 171(4):1741–1749, 2003.

[127] Fernando P. Polack, Stephen J. Thomas, Nicholas Kitchin, Judith Absalon, Alejandra Gurtman, Stephen Lockhart, John L. Perez, Gonzalo Pérez Marc, Edson D. Moreira, Cristiano Zerbini, Ruth Bailey, Kena A. Swanson, Satrajit Roychoudhury, Kenneth Koury, Ping Li, Warren V. Kalina, David Cooper, Robert W. Frenck, Laura L. Hammitt, Özlem Türeci, Haylene Nell, Axel Schaefer, Serhat Ünal, Dina B. Tresnan, Susan Mather, Philip R. Dormitzer, Uğur

Şahin, Kathrin U. Jansen, William C. Gruber, and C4591001 Clinical Trial Group. Safety and Efficacy of the BNT162b2 mRNA Covid-19 Vaccine. *The New England Journal of Medicine*, 383(27):2603–2615, December 2020.

[128] Angel Porgador, Jonathan W Yewdell, Yuping Deng, Jack R Bennink, and Ronald N Germain. Localization, Quantitation, and In Situ Detection of Specific Peptide–MHC Class I Complexes Using a Monoclonal Antibody. *Immunity*, 6(6):715–726, June 1997.

[129] Eric Reits, Alexander Griekspoor, Joost Neijssen, Tom Groothuis, Kees Jalink, Peter Van Veelen, Hans Janssen, Jero Calafat, Jan Wouter Drijfhout, and Jacques Neefjes. Peptide Diffusion, Protection, and Degradation in Nuclear and Cytoplasmic Compartments before Antigen Presentation by MHC Class I. *Immunity*, 18(1):97–108, January 2003.

[130] Birkir Reynisson, Bruno Alvarez, Sinu Paul, Bjoern Peters, and Morten Nielsen. 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, July 2020.

[131] J. Roelse, M. Grommé, F. Momburg, G. Hämmerling, and J. Neefjes. Trimming of TAP-translocated peptides in the endoplasmic reticulum and in the cytosol during recycling. *The Journal of Experimental Medicine*, 180(5):1591–1597, November 1994.

[132] Ankit Rohatgi. WebPlotDigitizer, September 2022.

[133] B. Sadasivan, P. J. Lehner, B. Ortmann, T. Spies, and P. Cresswell. Roles for calreticulin and a novel glycoprotein, tapasin, in the interaction of MHC class I molecules with TAP. *Immunity*, 5(2):103–114, August 1996.

[134] Ugur Sahin and Özlem Türeci. Personalized vaccines for cancer immunotherapy. *Science (New York, N.Y.)*, 359(6382):1355–1360, March 2018.

[135] A. Saltelli, S. Tarantola, and K. P.-S. Chan. A Quantitative Model-Independent

Method for Global Sensitivity Analysis of Model Output. *Technometrics*, 41(1):39–56, February 1999.

[136] Siranush Sarkizova, Susan Klaeger, Phuong M. Le, Letitia W. Li, Giacomo Oliveira, Hasmik Keshishian, Christina R. Hartigan, Wandi Zhang, David A. Braun, Keith L. Ligon, Pavan Bachireddy, Ioannis K. Zervantonakis, Jennifer M. Rosenbluth, Tamara Ouspenskaia, Travis Law, Sune Justesen, Jonathan Stevens, William J. Lane, Thomas Eisenhaure, Guang Lan Zhang, Karl R. Clauser, Nir Hacohen, Steven A. Carr, Catherine J. Wu, and Derin B. Keskin. A large peptidome dataset improves HLA class I epitope prediction across most of the human population. *Nature Biotechnology*, 38(2):199–209, February 2020.

[137] Loredana Saveanu, Oliver Carroll, Vivian Lindo, Margarita Del Val, Daniel Lopez, Yves Lepelletier, Fiona Greer, Lutz Schomburg, Doriana Fruci, Gabriele Niedermann, and Peter M. Van Endert. Concerted peptide trimming by human ERAP1 and ERAP2 aminopeptidase complexes in the endoplasmic reticulum. *Nature Immunology*, 6(7):689–697, 2005.

[138] Patricia Saxová, Søren Buus, Søren Brunak, and Can Keşmir. Predicting proteasomal cleavage sites: a comparison of available methods. *International Immunology*, 15(7):781–787, July 2003.

[139] Mark M. Schatz, Björn Peters, Nadja Akkad, Nina Ullrich, Alejandra Nacarino Martinez, Oliver Carroll, Sascha Bulik, Hans-Georg Rammensee, Peter Van Endert, Hermann-Georg Holzhütter, Stefan Tenzer, and Hansjörg Schild. Characterizing the N-Terminal Processing Motif of MHC Class I Ligands. *The Journal of Immunology*, 180(5):3210–3217, March 2008.

[140] Julien Schmidt, Angela R. Smith, Morgane Magnin, Julien Racle, Jason R. Devlin, Sara Bobisse, Julien Cesbron, Victor Bonnet, Santiago J. Carmona, Florian Huber, Giovanni Ciriello, Daniel E. Speiser, Michal Bassani-Sternberg, George Coukos, Brian M. Baker, Alexandre Harari, and David Gfeller. Prediction of neo-epitope immunogenicity reveals TCR recognition determinants

and provides insight into immunoediting. *Cell Reports Medicine*, 2(2):100194, 2021. Publisher: ElsevierCompany.

[141] T. N. Schumacher, D. V. Kantesaria, M. T. Heemels, P. G. Ashton-Rickardt, J. C. Shepherd, K. Fruh, Y. Yang, P. A. Peterson, S. Tonegawa, and H. L. Ploegh. Peptide length and sequence specificity of the mouse TAP1/TAP2 translocator. *The Journal of Experimental Medicine*, 179(2):533–540, February 1994.

[142] Sunesh Sethumadhavan, Marie Barth, Robbert M. Spaapen, Carla Schmidt, Simon Trowitzsch, and Robert Tampé. Viral immune evasins impact antigen presentation by allele-specific trapping of MHC I at the peptide-loading complex. *Scientific Reports*, 12(1):1516, January 2022.

[143] Xiaoshan M. Shao, Rohit Bhattacharya, Justin Huang, I. K. Ashok Sivakumar, Collin Tokheim, Lily Zheng, Dylan Hirsch, Benjamin Kaminow, Ashton Omdahl, Maria Bonsack, Angelika B. Riemer, Victor E. Velculescu, Valsamo Anagnostou, Kymberleigh A. Pagel, and Rachel Karchin. High-Throughput Prediction of MHC Class I and II Neoantigens with MHCnuggets. *Cancer Immunology Research*, 8(3):396–408, March 2020.

[144] Koen Smets, Brigitte Verdonk, and Elsa M. Jordaan. Evaluation of Performance Measures for SVR Hyperparameter Selection. In *2007 International Joint Conference on Neural Networks*, pages 637–642, Orlando, FL, USA, August 2007. IEEE. ISSN: 1098-7576.

[145] Ruey-Chyi Su and Richard G. Miller. Stability of Surface H-2Kb, H-2Db, and Peptide-Receptive H-2Kb on Splenocytes. *The Journal of Immunology*, 167(9):4869–4877, November 2001.

[146] S. Tenzer, B. Peters, S. Bulik, O. Schoor, C. Lemmel, M. M. Schatz, P.-M. Kloetzel, H.-G. Rammensee, H. Schild, and H.-G. Holzhütter. Modeling the MHC class I pathway by combining predictions of proteasomal cleavage, TAP transport and MHC class I binding. *Cellular and molecular life sciences: CMLS*, 62(9):1025–1037, May 2005.

[147] Kathrin Textoris-Taube, Clemens Cammann, Petra Henklein, Eylin Topfstedt, Frédéric Ebstein, Sarah Henze, Juliane Liepe, Fang Zhao, Dirk Schadendorf, Burkhardt Dahlmann, Wolfgang Uckert, Annette Paschen, Michele Mishto, and Ulrike Seifert. ER-aminopeptidase 1 determines the processing and presentation of an immunotherapy-relevant melanoma epitope. *European Journal of Immunology*, 50(2):270–283, 2020.

[148] The UniProt Consortium, Alex Bateman, Maria-Jesus Martin, Sandra Orchard, Michele Magrane, Shadab Ahmad, Emanuele Alpi, Emily H Bowler-Barnett, Ramona Britto, Hema Bye-A-Jee, Austra Cukura, Paul Denny, Tunca Dogan, ThankGod Ebenezer, Jun Fan, Penelope Garmiri, Leonardo Jose Da Costa Gonzales, Emma Hatton-Ellis, Abdulrahman Hussein, Alexandr Ignatchenko, Giuseppe Insana, Rizwan Ishtiaq, Vishal Joshi, Dushyanth Jyothi, Swaathi Kandasaamy, Antonia Lock, Aurelien Luciani, Marija Lugaric, Jie Luo, Yvonne Lussi, Alistair MacDougall, Fabio Madeira, Mahdi Mahmoudy, Alok Mishra, Katie Moulang, Andrew Nightingale, Sangya Pundir, Guoying Qi, Shriya Raj, Pedro Raposo, Daniel L Rice, Rabie Saidi, Rafael Santos, Elena Speretta, James Stephenson, Prabhat Totoo, Edward Turner, Nidhi Tyagi, Preethi Vasudev, Kate Warner, Xavier Watkins, Rossana Zaru, Hermann Zellner, Alan J Bridge, Lucila Aimo, Ghislaine Argoud-Puy, Andrea H Auchincloss, Kristian B Axelsen, Parit Bansal, Delphine Baratin, Teresa M Batista Neto, Marie-Claude Blatter, Jerven T Bolleman, Emmanuel Boutet, Lionel Breuza, Blanca Cabrera Gil, Cristina Casals-Casas, Kamal Chikh Echioukh, Elisabeth Coudert, Beatrice Cuche, Edouard De Castro, Anne Estreicher, Maria L Famiglietti, Marc Feuermann, Elisabeth Gasteiger, Pascale Gaudet, Sebastien Gehant, Vivienne Gerritsen, Arnaud Gos, Nadine Gruaz, Chantal Hulo, Nevila Hyka-Nouspikel, Florence Jungo, Arnaud Kerhornou, Philippe Le Mercier, Damien Lieberherr, Patrick Masson, Anne Morgat, Venkatesh Muthukrishnan, Salvo Paesano, Ivo Pedruzzi, Sandrine Pilbout, Lucille Pourcel, Sylvain Poux, Monica Pozzato, Manuela Pruess, Nicole Redaschi, Catherine Rivoire, Christian J A Sigrist, Karin Sonesson, Shyamala Sundaram, Cathy H Wu, Cecilia N Arighi, Leslie Arminski, Chuming Chen, Yongxing Chen, Hongzhan Huang,

Kati Laiho, Peter McGarvey, Darren A Natale, Karen Ross, C R Vinayaka, Qinghua Wang, Yuqi Wang, and Jian Zhang. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, January 2023.

[149] Alex Theodossis, Carole Guillonneau, Andrew Welland, Lauren K. Ely, Craig S. Clements, Nicholas A. Williamson, Andrew I. Webb, Jacqueline A. Wilce, Roger J. Mulder, Michelle A. Dunstone, Peter C. Doherty, James McCluskey, Anthony W. Purcell, Stephen J. Turner, and Jamie Rossjohn. Constraints within major histocompatibility complex class I restricted peptides: presentation and consequences for T-cell recognition. *Proceedings of the National Academy of Sciences of the United States of America*, 107(12):5534–5539, March 2010.

[150] Charles F. Towne, Ian A. York, Joost Neijssen, Margaret L. Karow, Andrew J. Murphy, David M. Valenzuela, George D. Yancopoulos, Jacques J. Neefjes, and Kenneth L. Rock. Leucine aminopeptidase is not essential for trimming peptides in the cytosol or generating epitopes for MHC class I antigen presentation. *Journal of Immunology (Baltimore, Md.: 1950)*, 175(10):6605–6614, November 2005.

[151] Charles F. Towne, Ian A. York, Joost Neijssen, Margaret L. Karow, Andrew J. Murphy, David M. Valenzuela, George D. Yancopoulos, Jacques J. Neefjes, and Kenneth L. Rock. Puromycin-sensitive aminopeptidase limits MHC class I presentation in dendritic cells but does not affect CD8 T cell responses during viral infections. *Journal of Immunology (Baltimore, Md.: 1950)*, 180(3):1704–1712, February 2008.

[152] Charles F. Towne, Ian A. York, Levi B. Watkin, John S. Lazo, and Kenneth L. Rock. Analysis of the role of bleomycin hydrolase in antigen presentation and the generation of CD8 T cell responses. *Journal of Immunology (Baltimore, Md.: 1950)*, 178(11):6923–6930, June 2007.

[153] Thomas Trolle and Morten Nielsen. NetTepi: an integrated method for the prediction of T cell epitopes. *Immunogenetics*, 66(7-8):449–456, August 2014.

[154] S. Uebel, T. H. Meyer, W. Kraas, S. Kienle, G. Jung, K. H. Wiesmuller, and R. Tampe. Requirements for peptide binding to the human transporter associated with antigen processing revealed by peptide scans and complex peptide libraries. *Journal of Biological Chemistry*, 270(31):18512–18516, 1995.

[155] Stephan Uebel, Wolfgang Kraas, Stefan Kienle, Karl Heinz Wiesmüller, Günther Jung, and Robert Tampé. Recognition principle of the TAP transporter disclosed by combinatorial peptide libraries. *Proceedings of the National Academy of Sciences of the United States of America*, 94(17):8976–8981, 1997.

[156] P M Van Endert, D Riganelli, G Greco, K Fleischhauer, J Sidney, A Sette, and J F Bach. The peptide-binding motif for the human transporter associated with antigen processing. *The Journal of experimental medicine*, 182(6):1883–1895, December 1995.

[157] Peter M. van Endert, Robert Tampé, Thomas H. Meyer, Roland Tisch, Jean François Bach, and Hugh O. McDevitt. A sequential model for peptide binding and transport by the transporters associated with antigen processing. *Immunity*, 1(6):491–500, 1994.

[158] Michel Van Kempen, Stephanie S. Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron L. M. Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with Foldseek. *Nature Biotechnology*, May 2023.

[159] Johan F Vansteenkiste, Byoung Chul Cho, Tonu Vanakesa, Tommaso De Pas, Marcin Zielinski, Moon Soo Kim, Jacek Jassem, Masahiro Yoshimura, Jubrail Dahabreh, Haruhiku Nakayama, Libor Havel, Haruhiko Kondo, Tetsuya Mitsudomi, Konstantinos Zarogoulidis, Oleg A Gladkov, Katalin Udud, Hirohito Tada, Hans Hoffman, Anders Bugge, Paul Taylor, Emilio Esteban Gonzalez, Mei Lin Liao, Jianxing He, Jean-Louis Pujol, Jamila Louahed, Muriel Debois,

Vincent Brichard, Channa Debruyne, Patrick Therasse, and Nasser Altorki. Efficacy of the MAGE-A3 cancer immunotherapeutic as adjuvant therapy in patients with resected MAGE-A3-positive non-small-cell lung cancer (MAGRIT): a randomised, double-blind, placebo-controlled, phase 3 trial. *The Lancet Oncology*, 17(6):822–835, June 2016.

[160] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. Van Der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul Van Mulbregt, SciPy 1.0 Contributors, Aditya Vijaykumar, Alessandro Pietro Bardelli, Alex Rothberg, Andreas Hilboll, Andreas Kloeckner, Anthony Scopatz, Antony Lee, Ariel Rokem, C. Nathan Woods, Chad Fulton, Charles Masson, Christian Häggström, Clark Fitzgerald, David A. Nicholson, David R. Hagen, Dmitrii V. Pasechnik, Emanuele Olivetti, Eric Martin, Eric Wieser, Fabrice Silva, Felix Lenders, Florian Wilhelm, G. Young, Gavin A. Price, Gert-Ludwig Ingold, Gregory E. Allen, Gregory R. Lee, Hervé Audren, Irvin Probst, Jörg P. Dietrich, Jacob Silterra, James T Webber, Janko Slavič, Joel Nothman, Johannes Buchner, Johannes Kulick, Johannes L. Schönberger, José Vinícius De Miranda Cardoso, Joscha Reimer, Joseph Harrington, Juan Luis Cano Rodríguez, Juan Nunez-Iglesias, Justin Kuczynski, Kevin Tritz, Martin Thoma, Matthew Newville, Matthias Kümmerer, Maximilian Bolingbroke, Michael Tartre, Mikhail Pak, Nathaniel J. Smith, Nikolai Nowaczyk, Nikolay Shebanov, Oleksandr Pavlyk, Per A. Brodtkorb, Perry Lee, Robert T. McGibbon, Roman Feldbauer, Sam Lewis, Sam Tygier, Scott Sievert, Sebastiano Vigna, Stefan Peterson, Surhud More, Tadeusz Pudlik, Takuya Oshima, Thomas J. Pingel, Thomas P. Robitaille, Thomas Spura, Thouis R. Jones, Tim Cera, Tim Leslie, Tiziano Zito, Tom Krauss, Utkarsh Upadhyay, Yaroslav O. Halchenko, and Yoshiki Vázquez-Baeza. SciPy 1.0: fundamental

algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272, March 2020.

[161] Merryn Voysey, Sue Ann Costa Clemens, Shabir A Madhi, Lily Y Weckx, Pedro M Folegatti, Parvinder K Aley, Brian Angus, Vicky L Baillie, Shaun L Barnabas, Qasim E Bhorat, Sagida Bibi, Carmen Briner, Paola Cicconi, Andrea M Collins, Rachel Colin-Jones, Clare L Cutland, Thomas C Darton, Keertan Dheda, Christopher J A Duncan, Katherine R W Emary, Katie J Ewer, Lee Fairlie, Saul N Faust, Shuo Feng, Daniela M Ferreira, Adam Finn, Anna L Goodman, Catherine M Green, Christopher A Green, Paul T Heath, Catherine Hill, Helen Hill, Ian Hirsch, Susanne H C Hodgson, Alane Izu, Susan Jackson, Daniel Jenkin, Carina C D Joe, Simon Kerridge, Anthonet Koen, Gaurav Kwatra, Rajeka Lazarus, Alison M Lawrie, Alice Lelliott, Vincenzo Libri, Patrick J Lillie, Raburn Mallory, Ana V A Mendes, Eveline P Milan, Angela M Minassian, Alastair McGregor, Hazel Morrison, Yama F Mujadidi, Anusha Nana, Peter J O'Reilly, Sherman D Padayachee, Ana Pittella, Emma Plested, Katrina M Pollock, Maheshi N Ramasamy, Sarah Rhead, Alexandre V Schwarzbold, Nisha Singh, Andrew Smith, Rinn Song, Matthew D Snape, Eduardo Sprinz, Rebecca K Sutherland, Richard Tarrant, Emma C Thomson, M Estée Török, Mark Toshner, David P J Turner, Johan Vekemans, Tonya L Villafana, Marion E E Watson, Christopher J Williams, Alexander D Douglas, Adrian V S Hill, Teresa Lambe, Sarah C Gilbert, Andrew J Pollard, Marites Aban, Fatola Abayomi, Kushala Abeyskera, Jeremy Aboagye, Matthew Adam, Kirsty Adams, James Adamson, Yemi A. Adelaja, Gbadebo Adewetan, Syed Adlou, Khatija Ahmed, Yasmeen Akhalwaya, Saajida Akhalwaya, Andrew Alcock, Aabidah Ali, Elizabeth R. Allen, Lauren Allen, Thamires C. D. S. C Almeida, Mariana P.S. Alves, Fabio Amorim, Foteini Andritsou, Rachel Anslow, Matthew Appleby, Edward H. Arbe-Barnes, Mark P. Ariaans, Beatriz Arns, Laiana Arruda, Paula Azi, Lorena Azi, Gavin Babbage, Catherine Bailey, Kenneth F. Baker, Megan Baker, Natalie Baker, Philip Baker, Lisa Baldwin, Ioana Baleanu, Danieli Bandeira, Anna Bara, Marcella A.S. Barbosa, Debbie Barker, Gavin D. Barlow, Eleanor Barnes, Andrew S. Barr, Jordan R.

Barrett, Jessica Barrett, Louise Bates, Alexander Batten, Kirsten Beadon, Emily Beales, Rebecca Beckley, Sandra Belij-Rammerstorfer, Jonathan Bell, Duncan Bellamy, Nancy Bellei, Sue Belton, Adam Berg, Laura Bermejo, Eleanor Berrie, Lisa Berry, Daniella Berzenyi, Amy Beveridge, Kevin R. Bewley, Helen Bexhell, Sutika Bhikha, Asad E. Bhorat, Zaheda E. Bhorat, Else Bijker, Geeta Birch, Sarah Birch, Adam Bird, Olivia Bird, Karen Bisnauthsing, Mustapha Bittaye, Katherine Blackstone, Luke Blackwell, Heather Bletchly, Caitlin L. Blundell, Susannah R. Blundell, Pritesh Bodalia, Bruno C. Boettger, Emma Bolam, Elena Boland, Daan Bormans, Nicola Borthwick, Francesca Bowring, Amy Boyd, Penny Bradley, Tanja Brenner, Phillip Brown, Claire Brown, Charlie Brown-O'Sullivan, Scott Bruce, Emily Brunt, Ruaridh Buchan, William Budd, Yusuf A. Bulbulia, Melanie Bull, Jamie Burbage, Hassan Burhan, Aileen Burn, Karen R. Buttigieg, Nicholas Byard, Ingrid Cabera Puig, Gloria Calderon, Anna Calvert, Susana Camara, Michelangelo Cao, Federica Cappuccini, João R. Cardoso, Melanie Carr, Miles W. Carroll, Andrew Carson-Stevens, Yasmin De M. Carvalho, José A.M. Carvalho, Helen R. Casey, Paul Cashen, Thais Castro, Lucia Carratala Castro, Katrina Cathie, Ana Cavey, José Cerbino-Neto, Jim Chadwick, David Chapman, Sue Charlton, Irina Chelysheva, Oliver Chester, Sunder Chita, Jee-Sun Cho, Liliana Cifuentes, Elizabeth Clark, Matthew Clark, Andrea Clarke, Elizabeth A. Clutterbuck, Sarah L.K. Collins, Christopher P. Conlon, Sean Connarty, Naomi Coombes, Cushla Cooper, Rachel Cooper, Lynne Cornelissen, Tumena Corrah, Catherine Cosgrove, Tony Cox, Wendy E.M. Crocker, Sarah Crosbie, Lorraine Cullen, Dan Cullen, Debora R.M.F. Cunha, Christina Cunningham, Fiona C. Cuthbertson, Suzete N. Farias Da Guarda, Larissa P. Da Silva, Brad E. Damratoski, Zsofia Danos, Maria T.D.C. Dantas, Paula Darroch, Mehreen S. Datoo, Chandrabali Datta, Malika Davids, Sarah L. Davies, Hannah Davies, Elizabeth Davis, Judith Davis, John Davis, Maristela M.D. De Nobrega, Lis Moreno De Oliveira Kalid, David Dearlove, Tesfaye Demissie, Amisha Desai, Stefania Di Marco, Claudio Di Maso, Maria I.S. Dinelli, Tanya Dinesh, Claire Docksey, Christina Dold, Tao Dong, Francesca R. Donnellan, Tannyth Dos San-

tos, Thainá G. Dos Santos, Erika Pachecho Dos Santos, Naomi Douglas, Charlotte Downing, Jonathan Drake, Rachael Drake-Brockman, Kimberley Driver, Ruth Drury, Susanna J. Dunachie, Benjamin S. Durham, Lidiana Dutra, Nicholas J.W. Easom, Samual Van Eck, Mandy Edwards, Nick J. Edwards, Omar M. El Muhanna, Sean C. Elias, Mike Elmore, Marcus English, Alisgair Esmail, Yakub Moosa Essack, Eoghan Farmer, Mutjaba Farooq, Madi Farrar, Leonard Farrugia, Beverley Faulkner, Sofiya Fedosyuk, Sally Felle, Shuo Feng, Carla Ferreira Da Silva, Samantha Field, Richard Fisher, Amy Flaxman, James Fletcher, Hazel Fofie, Henry Fok, Karen J. Ford, Jamie Fowler, Pedro H.A. Fraiman, Emma Francis, Marilia M. Franco, John Frater, Marilúcia S.M. Freire, Samantha H. Fry, Sabrina Fudge, Julie Furze, Michelle Fuskova, Pablo Galian-Rubio, Eva Galiza, Harriet Garlant, Madita Gavrila, Ailsa Geddes, Karyna A. Gibbons, Ciaran Gilbride, Hardeep Gill, Sharon Glynn, Kerry Godwin, Karishma Gokani, Ursula Carvalho Goldoni, Maria Goncalves, Isabela G.S. Gonzalez, Jayne Goodwin, Amina Goondiwala, Katherine Gordon-Quayle, Giacomo Gorini, Janet Grab, Lara Gracie, Melanie Greenland, Nicola Greenwood, Johann Greffrath, Marisa M. Groenewald, Leonardo Grossi, Gaurav Gupta, Mark Hackett, Bassam Hallis, Mainga Hamaluba, Elizabeth Hamilton, Joseph Hamlyn, Daniel Hammersley, Aidan T. Hanrath, Brama Hanumunthadu, Stephanie A. Harris, Clair Harris, Tara Harris, Thomas D. Harrison, Daisy Harrison, Thomas C. Hart, Birgit Hartnell, Shadin Hassan, John Haughney, Sophia Hawkins, Jodie Hay, Ian Head, John Henry, Macarena Hermosin Herrera, David B. Hettle, Jennifer Hill, Gina Hodges, Elizea Horne, Mimi M. Hou, Catherine Houlihan, Elizabeth Howe, Nicola Howell, Jonathan Humphreys, Holly E. Humphries, Katrina Hurley, Claire Huson, Angela Hyder-Wright, Catherine Hyams, Sabina Ikram, Alka Ishwarbhai, Monica Ivan, Poppy Iveson, Vidyashankara Iyer, Frederic Jackson, Jeanne De Jager, Shameem Jaumdally, Helen Jeffers, Natasha Jesudason, Bryony Jones, Kathryn Jones, Elizabeth Jones, Christopher Jones, Marianna Rocha Jorge, Aylin Jose, Amar Joshi, Eduardo A.M.S. Júnior, Joanne Kadziola, Reshma Kailath, Faeeza Kana, Konstantinos Karampatsas, Mwila Kasanyinga, Jade Keen, Elizabeth J.

Kelly, Dearbhla M. Kelly, Debbie Kelly, Sarah Kelly, David Kerr, Renato De Ávila Kfouri, Liaquat Khan, Baktash Khozoee, Sarah Kidd, Annabel Killen, Jasmin Kinch, Patrick Kinch, Lloyd D.W. King, Thomas B. King, Lucy Kingham, Paul Klenerman, Francesca Knapper, Julian C. Knight, Daniel Knott, Stanislava Koleva, Matilda Lang, Gail Lang, Colin W. Larkworthy, Jessica P.J. Larwood, Rebecca Law, Erica M. Lazarus, Amanda Leach, Emily A. Lees, Nana-Marie Lemm, Alvaro Lessa, Stephanie Leung, Yuanyuan Li, Amelia M. Lias, Kostas Liatsikos, Aline Linder, Samuel Lipworth, Shuchang Liu, Xinxue Liu, Adam Lloyd, Stephanie Lloyd, Lisa Loew, Raquel Lopez Ramon, Leandro Lora, Vicki Lowthorpe, Kleber Luz, Jonathan C. MacDonald, Gordon MacGregor, Meera Madhavan, David O. Mainwaring, Edson Makambwa, Rebecca Makinson, Mookho Malahleha, Ross Malamatsho, Garry Mallett, Kushal Mansatta, Takalani Maoko, Katlego Mapetla, Natalie G. Marchevsky, Spyridoula Marinou, Emma Marlow, Gabriela N. Marques, Paula Marriott, Richard P. Marshall, Julia L. Marshall, Flávia J. Martins, Masebole Masenya, Mduduzi Masilela, Shauna K. Masters, Moncy Mathew, Hosea Matlebjane, Kedidimetse Matshidiso, Olga Mazur, Andrea Mazzella, Hugh McCaughan, Joanne McEwan, Joanna McGlashan, Lorna McInroy, Zoe McIntyre, Daniela McLenaghan, Nicky McRobert, Steve McSwiggan, Clare Megson, Savviz Mehdipour, Wilma Meijs, Renata N.Á. Mendonça, Alexander J. Mentzer, Neginsadat Mirtorabi, Celia Mitton, Sibusiso Mnyakeni, Fiona Moghaddas, Kgaogelo Molapo, Mapule Moloi, Maria Moore, M. Isabel Moraes-Pinto, Marni Moran, Ella Morey, Róisín Morgans, Susan Morris, Sheila Morris, Helen C. Morris, Franca Morselli, Gertraud Morshead, Richard Morter, Lynelle Mottal, Andrew Moultrie, Nathifa Moya, Mushiya Mpelembue, Sibekezelo Msomi, Yvonne Mugodi, Ekta Mukhopadhyay, Jilly Muller, Alasdair Munro, Claire Munro, Sarah Murphy, Philomena Mweu, Celia Hatsuko Myasaki, Gurudutt Naik, Kush Naker, Eleni Nastouli, Abida Nazir, Bongani Ndlovu, Fabio Neffa, Cecilia Njenga, Helena Noal, Andrés Noé, Gabrielle Novaes, Fay L. Nugent, Géssika Nunes, Katie O'Brien, Daniel O'Connor, Miranda Odam, Suzette Oelofse, Blanche Oguti, Victoria Olchawski, Neil J. Oldfield, Marianne G.

Oliveira, Catarina Oliveira, Angela Oosthuizen, Paula O'Reilly, Piper Osborne, David R.J. Owen, Lydia Owen, Daniel Owens, Nelly Owino, Mihaela Pacurar, Brenda V.B. Paiva, Edna M.F. Palhares, Susan Palmer, Sivapriyai Parkinson, Helena M.R.T. Parracho, Karen Parsons, Dipak Patel, Bhumika Patel, Faeezah Patel, Kelly Patel, Maia Patrick-Smith, Ruth O. Payne, Yanchun Peng, Elizabeth J. Penn, Anna Pennington, Marco Polo Peralta Alvarez, James Perring, Nicola Perry, Rubeshan Perumal, Sahir Petkar, Tricia Philip, Daniel J. Phillips, Jennifer Phillips, Mary Kgomotso Phohu, Lorinda Pickup, Sonja Pieterse, Jo Piper, Dimitra Pipini, Mary Plank, Joan Du Plessis, Samuel Pollard, Jennifer Pooley, Anil Pooran, Ian Poulton, Claire Powers, Fernando B. Presa, David A. Price, Vivien Price, Marcelo Primeira, Pamela C. Proud, Samuel Provstgaard-Morys, Sophie Pueschel, David Pulido, Sheena Quaid, Ria Rabara, Alexandra Radford, Kajal Radia, Durga Rajapaska, Thurkka Rajeswaran, Alberto San Francisco Ramos, Fernando Ramos Lopez, Tommy Rampling, Jade Rand, Helen Ratcliffe, Tom Rawlinson, David Rea, Byron Rees, Jesús Reiné, Mila Resuello-Dauti, Emilia Reyes Pabon, Carla M. Ribiero, Marivic Ricamara, Alex Richter, Neil Ritchie, Adam J. Ritchie, Alexander J. Robbins, Hannah Roberts, Ryan E. Robinson, Hannah Robinson, Talita T. Rocchetti, Beatriz Pinho Rocha, Sophie Roche, Christine Rollier, Louisa Rose, Amy L. Ross Russell, Lindie Rossouw, Simon Royal, Indra Rudiansyah, Sarah Ruiz, Stephen Saich, Claudia Sala, Jessica Sale, Ahmed M. Salman, Natalia Salvador, Stephannie Salvador, Milla Sampaio, Annette D. Samson, Amada Sanchez-Gonzalez, Helen Sanders, Katherine Sanders, Erika Santos, Mayara F.S. Santos Guerra, Iman Satti, Jack E. Saunders, Caroline Saunders, Aakifah Sayed, Ina Schim Van Der Loeff, Annina B. Schmid, Ella Schofield, Gavin Screaton, Samiullah Seddiqi, Rameswara R. Segireddy, Roberta Senger, Sonia Serrano, Rajiv Shah, Imam Shaik, Hannah E. Sharpe, Katherine Sharrocks, Robert Shaw, Adam Shea, Amy Shepherd, James G. Shepherd, Farah Shiham, Emad Sidhom, Sarah E. Silk, Antonio Carlos Da Silva Moraes, Gilberto Silva-Junior, Laura Silva-Reyes, Anderson D. Silveira, Mariana B.V. Silveira, Jaisi Sinha, Donal T. Skelly, Daniel C. Smith, Nick Smith, Holly E. Smith, David J. Smith,

Catherine C. Smith, Airanuédida Soares, Tiago Soares, Carla Solórzano, Guilherme L. Sorio, Kim Sorley, Tiffany Sosa-Rodriguez, Cinthia M.C.D.L. Souza, Bruno S.D.F. Souza, Alessandra R. Souza, Alexandra J. Spencer, Fernanda Spina, Louise Spoors, Lizzie Stafford, Imogen Stamford, Igor Starinskij, Ricardo Stein, Jill Steven, Lisa Stockdale, Lisa V. Stockwell, Louise H. Strickland, Arabella C. Stuart, Ann Sturdy, Natalina Sutton, Anna Szigeti, Abdessamad Tahiri-Alaoui, Rachel Tanner, Carol Taoushanis, Alexander W. Tarr, Keja Taylor, Ursula Taylor, Iona Jennifer Taylor, Justin Taylor, Rebecca Te Water Naude, Yrene Themistocleous, Andreas Themistocleous, Merin Thomas, Kelly Thomas, Tonia M. Thomas, Asha Thombrayil, Fawziyah Thompson, Amber Thompson, Kevin Thompson, Ameeka Thompson, Julia Thomson, Viv Thornton-Jones, Patrick J. Tighe, Lygia Accioly Tinoco, Gerlynn Tiongson, Bonolo Tladinyane, Michele Tomasicchio, Adriana Tomic, Susan Tonks, James Towner, Nguyen Tran, Julia Tree, Gerry Trillana, Charlotte Trinham, Rose Trivett, Adam Truby, Betty Lebogang Tsheko, Aadil Turabi, Richard Turner, Cheryl Turner, Marta Ulaszewska, Benjamin R. Underwood, Rachel Varughese, Dennis Verbart, Marije Verheul, Iason Vichos, Taiane Vieira, Claire S. Waddington, Laura Walker, Erica Wallis, Matthew Wand, Deborah Warbick, Theresa Wardell, George Warimwe, Sarah C. Warren, Bridget Watkins, Ekaterina Watson, Stewart Webb, Alice Webb-Bridges, Angela Webster, Jessica Welch, Jeanette Wells, Alison West, Caroline White, Rachel White, Paul Williams, Rachel L. Williams, Rebecca Winslow, Mark Woodyer, Andrew T. Worth, Danny Wright, Marzena Wroblewska, Andy Yao, Rafael Zimmer, Dalila Zizi, and Peter Zuidewind. Safety and efficacy of the ChAdOx1 nCoV-19 vaccine (AZD1222) against SARS-CoV-2: an interim analysis of four randomised controlled trials in Brazil, South Africa, and the UK. *The Lancet*, 397(10269):99–111, January 2021.

[162] Yat-tsai Richie Wan, Zeynep Koşaloğlu-Yalçın, Bjoern Peters, and Morten Nielsen. A large-scale study of peptide features defining immunogenicity of cancer neo-epitopes. *NAR Cancer*, 6(1):zcae002, January 2024.

[163] Bo Wang and Eric R. Gamazon. Modeling mutational effects on biochemical phenotypes using convolutional neural networks: application to SARS-CoV-2. *iScience*, 25(7):104500, July 2022.

[164] Jamie Ware, Patrick McIntyre, Kristopher Clark, Carmen Tong, Jason Shiers, Elisa Lori, Camila De Almeida, Emma Reeves, Henry Leonard, Alihussein Remtulla, Michael Ford, Nicola Ternette, Fergus Poynton, Edd James, Lesley Young, Martin Quibell, and Peter I. Joyce. Abstract 5551: Potent oral ERAP1 inhibitors modify the immunopeptidome *in vivo* and are novel immunotherapy agents. *Cancer Research*, 80(16_Supplement):5551–5551, August 2020.

[165] Benjamin R Weeder, Mary A Wood, Ellysia Li, Abhinav Nellore, and Reid F Thompson. pepsickle rapidly and accurately predicts proteasomal cleavage sites for improved neoantigen identification. *Bioinformatics*, 37(21):3723–3733, November 2021.

[166] Mirjana Weimershaus, Irini Evnouchidou, Loredana Saveanu, and Peter Van Endert. Peptidases trimming MHC class I ligands. *Current Opinion in Immunology*, 25(1):90–96, February 2013.

[167] Anthony P. Williams, Chen Au Peh, Anthony W. Purcell, James McCluskey, and Tim Elliott. Optimization of the MHC class I peptide cargo is dependent on tapasin. *Immunity*, 16(4):509–520, 2002.

[168] Ting Wu, Jing Guan, Andreas Handel, David C. Tscharke, John Sidney, Alessandro Sette, Linda M. Wakim, Xavier Y. X. Sng, Paul G. Thomas, Nathan P. Croft, Anthony W. Purcell, and Nicole L. La Gruta. Quantification of epitope abundance reveals the effect of direct and cross-presentation on influenza CTL responses. *Nature Communications*, 10(1):2846, June 2019.

[169] Jonathan W. Yewdell, Eric Reits, and Jacques Neefjes. Making sense of mass destruction: Quantitating MHC class I antigen presentation. *Nature Reviews Immunology*, 3(12):952–961, 2003.

[170] Ian A. York, Shih-Chung Chang, Tomo Saric, Jennifer A. Keys, Janice M. Favreau, Alfred L. Goldberg, and Kenneth L. Rock. The ER aminopeptidase

ERAP1 enhances or limits antigen presentation by trimming epitopes to 8–9 residues. *Nature Immunology*, 3(12):1177–1184, December 2002.

[171] Efthalia Zervoudi, Athanasios Papakyriakou, Dimitra Georgiadou, Irini Evnouchidou, Anna Gajda, Marcin Poreba, Guy S. Salvesen, Marcin Drag, Akira Hattori, Luc Swevers, Dionisios Vourloumis, and Efstratios Stratikos. Probing the S1 specificity pocket of the aminopeptidases that generate antigenic peptides. *Biochemical Journal*, 435(2):411–420, April 2011.

[172] Xue Zhang, Jingcheng Wu, Joseph Baeza, Katie Gu, Yichun Zheng, Shuqing Chen, and Zhan Zhou. DeepTAP: An RNN-based method of TAP-binding peptide prediction in the selection of tumor neoantigens. *Computers in Biology and Medicine*, 164:107247, September 2023.

[173] Weilong Zhao and Xinwei Sher. Systematically benchmarking peptide-MHC binding predictors: From synthetic to naturally processed epitopes. *PLOS Computational Biology*, 14(11):e1006457, November 2018.

[174] Caifang Zheng, Weihao Shao, Xiaorui Chen, Bowen Zhang, Gaili Wang, and Weidong Zhang. Real-world effectiveness of COVID-19 vaccines: a literature review and meta-analysis. *International Journal of Infectious Diseases*, 114:252–260, January 2022.

[175] Ingo Ziegler, Bolei Ma, Bernd Bischl, Emilio Dorigatti, and Benjamin Schubert. Proteasomal cleavage prediction: state-of-the-art and future directions. preprint, Bioinformatics, July 2023.