Improving Neoantigen Prioritization Methods for Personalized Cancer Vaccines

Huiming Xia

Follow this and additional works at: https://openscholarship.wustl.edu/art_sci_etds

Recommended Citation
https://openscholarship.wustl.edu/art_sci_etds/2922

This Dissertation is brought to you for free and open access by the Arts & Sciences at Washington University Open Scholarship. It has been accepted for inclusion in Arts & Sciences Electronic Theses and Dissertations by an authorized administrator of Washington University Open Scholarship. For more information, please contact digital@wumail.wustl.edu.
Improving Neoantigen Prioritization Methods for Personalized Cancer Vaccines
by
Huiming Xia

A dissertation presented to
Washington University in St. Louis
in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

May 2023
St. Louis, Missouri
# Table of Contents

List of Figures v  
List of Tables vii  
Acknowledgments viii  
Abstract of the dissertation xi  

## Chapter 1: Introduction  
1.1 Preamble 1  
1.2 Introduction to neoantigens 1  
1.3 Neoantigen identification 4  
1.4 HLA typing, expression, and mutation analysis 8  
1.5 Predicting peptide processing 10  
1.6 MHC binding prediction 12  
1.7 Neoantigen prioritization and vaccine design pipelines 15  
1.8 Peptide creation, delivery mechanisms and related analysis considerations for vaccine design 18  
1.9 T cell recognition, TCR profiling, and immune cell profiling to evaluate response 20  
1.10 Conclusions and future directions 23  
1.11 Acknowledgements 26  
1.12 Author contributions 26

## Chapter 2: Computational prediction of MHC anchor locations guides neoantigen identification and prioritization 27  
2.1 Preamble 27  
2.2 Introduction 27  
2.3 Results 32  
   2.3.1 Computational and quantitative prediction of HLA-specific anchor positions 32  
   2.3.2 Prediction results show distinct patterns of HLA anchor locations 35  
   2.3.3 Protein structural analysis confirms predicted anchor results 43  
   2.3.4 Experimental validation shows similar anchor trends as our prediction across distinct HLA alleles 48  
   2.3.5 Neoantigen prioritization results are influenced by accounting for anchor locations 55  
2.4 Methods 61  
   2.4.1 Input data for identifying strong binding seed peptides for anchor site prediction 61  
   2.4.2 Computational prediction of anchor site positions for 328 HLA alleles 62  
   2.4.3 Evaluating different seed peptide sources and their impact on the anchor patterns predicted 64  
   2.3.4 Bias analysis for HLA allele-specific anchor patterns 64
Table 1.2 Key analysis considerations and practical guidance for clinical neoantigen workflows 117
Table 1.3 MHC class I binding algorithm comparison 135
Appendix 2: Chapter 2 Supplemental Materials 139
References 141
List of Figures

Figure 1.1. Overview of bioinformatic characterization of neoantigens 2
Figure 2.1. Anchor and mutation position scenarios at the MHC-peptide-TCR interface 29
Figure 2.2. Distribution of peptides collected per allele across 328 HLA alleles, split by peptide length 32
Figure 2.3. Overview of computational workflow for anchor position prediction 34
Figure 2.4. Saturation analysis for evaluating subsample size of peptides needed for simulation analysis 35
Figure 2.5. Hierarchical clustering of anchor prediction scores across all 9-mer peptides 36
Figure 2.6. Hierarchical clustering of anchor prediction scores across all 8, 10, and 11-mer peptides assembled 37
Figure 2.7. Comparison of anchor pattern across different seed peptide sources using HLA-A*02:01 38
Figure 2.8. Analysis of potential for supporting algorithm bias across 328 HLA alleles 41
Figure 2.9. Orthogonal validation using protein crystallography structures 44
Figure 2.10. Analysis of crystallography data for HLA-B*08:01 and 9-mer peptides 46
Figure 2.11. Distribution of Spearman correlation values comparing anchor predictions and peptide-HLA/TCR distance measurements 47
Figure 2.12. Experimental Validation of anchor pattern for HLA-B*07:02 49
Figure 2.13. Additional experimental validation data for the predicted HLA-B*08:01 anchor pattern 50
Figure 2.14. Additional experimental validation data for HLA-A*23:01 and HLA-A*68:01 51
Figure 2.15. Summary of experimental validation results across all HLA allele-peptide combinations 53
Figure 2.16. Breakdown of predicted binding affinity values versus measured binding affinity by individual algorithms 54
Figure 2.17. Correlation Scores between individual MHC binding algorithm predictions and measured binding affinities across HLA alleles 55
Figure 2.18. Impact of anchor position information on neoantigen prioritization decisions 56
Figure 2.19. Distribution of anchor scenario categories for neoantigen candidates from 923 tumor-HLA paired samples 58
Figure 2.20. Patient-level analysis for impact of anchor considerations on neoantigen prioritization 60
Figure 3.1. Overall process of neoantigen generation and presentation from tumor specific mutations 81
Figure 3.2. Overview of example workflow for prioritizing neoantigens using pVACview 85
Figure 4.1. Comparison of individual features between positive and negative validation dataset 101
Figure 4.2. Comparison of predicted IC50 binding affinity and percentile values across different algorithms and validation outcome groups
List of Tables

Table 4.1. Summary of neoantigen related studies collected with T-cell validation datasets 98
Table 4.2. Comparison across different algorithm predictions between positive and negative validation outcome datasets 102
Table 4.3. Summary of neoantigen candidates produced from in-house immunogenomics pipeline compared to published candidates 104
Acknowledgments

It still feels so surreal as I am writing this acknowledgement section and reflecting on my journey towards completing my doctoral dissertation. It has been an incredible experience, filled with ups and downs, and I couldn't have possibly made it this far without the help and support of so many incredible people.

To Malachi and Obi, I am so thankful and appreciative of your support through my graduate school years. When I first started rotating in the lab I had little to no background in genomic sequencing analysis and was quite intimidated and unsure of my abilities. However, you always took the time to answer my countless questions and never doubted my capabilities. I feel incredibly lucky to have you guys as advisor and mentor and I will always be grateful for everything you have taught me.

To my committee members, Bob, Todd, Will and Josh, thank you so much for your time, effort, and expertise in serving on my thesis committee. Your feedback, guidance, and expertise in your respective fields have been invaluable in shaping my research and helping me grow as a scholar. I feel so grateful to have had the opportunity to work with such a dedicated and knowledgeable committee.

To past and current members of the Griffith lab, thank you for making my time in the lab such a wonderful experience. I am not only grateful for the friendships and the memories we have created but also all the scientific discussion and insights you guys have given me. Special thanks to all my cubby-mates Megan, Sharon, Kelsy, Cody, Zach, Felicia, Susanna, Chris, Kartik, My and Mariam. Six years of graduate school was honestly the longest period of education I’ve spent in one place and I’m so glad I got to spend it in such a supportive and loving lab environment.
To my day ones, Rose, Jasmine, Jhullian, Matt, Paige, Victoria, Mickey, Naomi and Kellan, I still remember orientation week when we met 6 years ago and watched the solar eclipse together on Rose’s birthday. I can’t emphasize enough how much this support network has meant to me when I’ve been away from family. To my girls, Snow, Julia and Joyce, I honestly feel so lucky to have met you all during grad school. Thank you for being such amazing friends and for making my grad school experience so unforgettable. To all the wonderful people I’ve met and friendships formed throughout my grad school journey, the bonds we forged will always hold a special place in my heart.

Last but not least, to my parents, Jun Xia and Lei Ming, I couldn’t possibly be here today without your support and guidance along the way. You are the full embodiment of resilience and perseverance and you continue to inspire me everyday. I am so grateful for the sacrifices that you have made to provide me with the best possible opportunities and for your unwavering belief in me even when I continue to doubt myself. I will always be grateful for everything that you have done for me.

Huiming Xia

Washington University in St. Louis

May 2023
Dedicated to my parents Jun & Lei.

Thank you for all your support.
ABSTRACT OF THE DISSERTATION

Improving neoantigen prioritization methods for personalized cancer vaccines

by

Huiming Xia

Doctor of Philosophy in Biology and Biomedical Sciences
Computational and Systems Biology
Washington University in St. Louis, 2023
Professor Malachi Griffith, Chair

Neoantigens are novel peptide sequences resulting from sources such as somatic mutations in tumors. Upon loading onto major histocompatibility complex (MHC) molecules, they can trigger recognition by T cells. Neoantigen targeting therapies including personalized vaccines have shown promise in the treatment of cancers, particularly when used in combination with checkpoint blockade therapy. Thus, accurate identification and prioritization of neoantigens is critical for conducting neoantigen-based clinical trials, predicting treatment response, and understanding mechanisms of resistance. With the advent of whole exome and RNA sequencing technologies, researchers and clinicians are now able to computationally predict neoantigens based on patient-specific mutation information. However, numerous factors must be considered when prioritizing neoantigens for use in personalized therapies, including but not limited to binding affinity, mutation location, mutant allele expression, allele-specific anchor locations. Complexities such as alternative transcript annotations, multiple algorithm prediction scores and variable peptide lengths/registers all potentially impact the neoantigen selection process. While there has been a rapid development of computational tools that attempt to account for these complexities, such as pVACtools, there remains considerable room for improvement of both the
underlying algorithms as well as how the data is integrated and visualized. To help address these issues, Chapter 2 of this thesis describes a computational pipeline for predicting allele-specific anchor locations. We computationally predicted anchor positions for different peptide lengths for 328 common HLA alleles and identified unique anchoring patterns among them. Analysis of 923 tumor samples shows that 6-38% of neoantigen candidates are potentially misclassified and can be rescued using allele-specific knowledge of anchor positions. A subset of anchor results was orthogonally validated using protein crystallography structures and representative anchor trends were experimentally validated using peptide-MHC stability assays and competition binding assays. Chapter 3 of this thesis describes pVACview, a novel interactive tool designed to aid in the prioritization and selection of neoantigen candidates for personalized neoantigen therapies including cancer vaccines. While computational pipelines generate numerous algorithmic predictions for neoantigen characterization, results from these pipelines are difficult to navigate and require extensive knowledge of the underlying tools for accurate interpretation. pVACview has a user-friendly and intuitive interface where users can upload, explore, select and export their neoantigen candidates. The tool allows users to visualize candidates across three different levels, including variant, transcript and peptide information for prioritization with greater efficiency and accuracy. It is also available as part of the pVACtools pipeline. To account for the different levels of complexity discussed in Chapters 2 and 3, Chapter 4 describes an ongoing effort to generate a standardized dataset of neoantigen features using experimentally validated neoantigens collected from publications. By acquiring raw sequencing datasets and running them through a formalized pipeline, we aim to generate a comprehensive collection of neoantigen features for future training and testing of various machine learning models. Overall, these chapters describe various efforts to improve current algorithms and methods pertaining to the
identification, prioritization and selection process of neoantigen candidates. We hope our findings will help formalize, streamline and improve the identification process for relevant clinical studies involving personalized cancer vaccines.
Chapter 1: Introduction

1.1 Preamble

The following chapter has been published as a review article:


* Authors Contributed Equally

As an author of the published manuscript, and in compliance with the editorial policies at Genome Medicine, the cited publication is included in full in the following chapter.

1.2 Introduction to neoantigens

The adaptive immune system has inherent antitumor properties capable of inducing tumor-specific cell death \(^1,2\). CD8+ and CD4+ T cells, two immune cell types critical to this process, recognize antigens bound by class I and II major histocompatibility complexes (MHC) on the cell surface, respectively. After antigen recognition, T cells have the ability to signal growth arrest and cell death to tumor cells displaying the antigen, as well as release paracrine signals to propagate an anti-tumor response. Neoantigens are specifically defined here as peptides derived from somatic mutations that provide an avenue for tumor-specific immune cell recognition and are important targets for cancer immunotherapies \(^3\, ^5\). Studies have shown that in addition to tumor mutational burden (TMB), high neoantigen burden can be a predictor of response to immune checkpoint blockade (ICB) therapy \(^6\, ^7\). This treatment strategy targets the signaling
pathways that suppress antitumor immune responses, allowing for the activation of neoantigen-specific T cells and promoting immune-mediated tumor cell death. Therefore, accurate neoantigen prediction is vital for the success of personalized vaccines as well as prioritizing candidates underlying the mechanism of response to ICB. These approaches have great therapeutic potential because neoantigen-specific T cells should not be susceptible to central tolerance.

With the advent of next generation sequencing (NGS), researchers can now rapidly sequence a patient’s DNA and RNA and perform analysis of these data to computationally predict neoantigens. This process requires several steps, each involving use of bioinformatics tools and complex analytical pipelines (Figure 1.1 & Table 1.1). Matched tumor/normal DNA
Figure 1.1. Overview of bioinformatic characterization of neoantigens. Major analysis steps in a comprehensive neoantigen characterization workflow are depicted in a simplified form. For each component, critical concepts and analysis considerations are indicated. Specific exemplar bioinformatics tools involved in each step are indicated in italics. Starting at the top left, patient sequences are analyzed to determine HLA types and predict corresponding MHC complexes for each tumor. Somatic variants of various types including SNVs (blue), deletions (red), insertions (green) and fusions (pink) are detected and corresponding peptide sequences are analyzed with respect to their predicted expression, processing and ability to bind the patients MHC complexes. Candidates are then selected for vaccine design and additional analyses are performed to assess T cell response.

sequencing data are processed and analyzed to call somatic mutations of various types. Human leukocyte antigen (HLA) haplotyping is performed to determine a patient’s HLA alleles and corresponding MHC complexes. Finally, RNA-seq data is used to quantify gene and transcript expression, which can verify variant expression prior to neoantigen prediction. Multiple pipelines exist to identify candidate neoantigens with high binding affinities to MHC class I or II. Additional steps are required to subsequently prioritize them for clinical use in personalized vaccines and address manufacturing and delivery issues.

The general concept of neoantigens and their role in personalized immunotherapies have been extensively reviewed elsewhere. Though experimental methods exist to assess neoantigens (e.g. mass spectrometry (MS)), the focus of this review is a comprehensive survey of computational approaches (tools, databases, and pipelines) for neoantigen characterization. While the ultimate goal is to discover neoepitopes, the part of the neoantigen recognized and bound by T cells, current workflows are largely focused on predicting peptide-MHC (pMHC) binding antigens with limited prediction of recognition by T cells or therapeutic potential. We have been particularly inspired by the use of computational approaches in human clinical trials involving personalized neoantigen vaccines alone or in combination with ICB. While a rapid expansion of the number and diversity of these trials has occurred over the past few years, there is limited community consensus on approaches for neoantigen characterization. Adoption of
standards for accurate identification of neoantigens and reporting of their features will be critical for interpretation of results from early stage trials and optimization of future trials. This review is focused on human clinical data; however, neoantigen characterization work involving model organisms (e.g. mice) is critical to advance the field and many of the tools/approaches described herein may be applied to these with appropriate modifications. In addition to describing emerging best practices, we will highlight current limitations and critical areas for improvement of the computational approaches needed to understand the immunogenicity of neoantigens.

1.3 Neoantigen identification

Two types of antigens that can induce an antitumor response are tumor-specific antigens, or neoantigens, and tumor-associated antigens (TAA). Neoantigens contain altered amino acid sequences that result from non-silent somatic mutations, while TAAs, which may originate from endogenous proteins or retroviruses, are selectively expressed or overexpressed by tumor cells but may also be expressed by non-tumor cell populations [13]. This review will focus on the detection and selection of neoantigens, but many analytical steps can apply to other antigen types. Considerations such as sample type (fresh frozen (FF), formalin-fixed paraffin-embedded (FFPE) tissue, ctDNA), tumor type (solid, blood), biopsy site, and sequencing approach (DNA, RNA, targeted sequencing) can impact somatic variant detection and interpretation and should be taken into account for data processing and downstream analysis [13–16]. In addition, tumors exhibiting high intratumoral heterogeneity can require alternative methods such as collecting multiple biopsies per tumor [17].

Somatic variant callers identify single nucleotide variants (SNVs) from tumor and matched non-tumor DNA sequence data such as whole genome, or more commonly, whole exome sequencing (WES) data [18]. Three common limitations to SNV calling, low frequency
variant detection, distinguishing germline variants from tumor in normal contamination, and removing sequencing artifacts, have been addressed by the following variant callers. MuTect2\textsuperscript{19} and Strelka\textsuperscript{20} have high sensitivity to detect SNVs at low allele fractions, thus enabling accurate subclonal variant detection. VarScan2\textsuperscript{21} and SomaticSniper\textsuperscript{22} require higher allele fractions for recognizing variants but can improve performance in cases of tumor in normal contamination\textsuperscript{23,24}. MuTect2 can further exclude sequencing or alignment artifacts by implementing a panel-of-normals file, containing false positives detected across normal samples. Running multiple variant calling algorithms simultaneously is recommended and can result in higher detection accuracy. For example, Callari et al. achieved 17.1\% higher sensitivity without increasing the false positive rate by intersecting a single variant callers’ results from multiple alignment pipelines and combining the intersected results from two callers, MuTect2 and Strelka, to achieve a final consensus\textsuperscript{25}. This list of variant callers is not exhaustive (Table 1.1 for additional options) and high quality pipelines using different combinations are certainly possible. Regardless of the combination of callers used, manual review of matched tumor/normal samples in Integrative Genomics Viewer (IGV)\textsuperscript{26} with a documented standard operating procedure is recommended to further reduce false positives\textsuperscript{27}. In addition to IGV, targeted sequencing approaches such as custom capture reagents can be utilized for further variant validation.

Recently, neoantigen vaccine trials for melanoma demonstrated that SNV-derived neoantigens can expand T cell populations\textsuperscript{28} and induce disease regression\textsuperscript{29,30}. However, recent studies have also increased appreciation for diverse neoantigen sources beyond simple SNVs to include short insertions and deletions (indels)\textsuperscript{31}, fusions\textsuperscript{32,33}, intron retention\textsuperscript{34}, non-coding expressed regions\textsuperscript{35}, exon-exon junction epitopes\textsuperscript{36}, B cell receptor (BCR) and T cell receptor (TCR) sequences for B and T cell malignancies respectively\textsuperscript{37}, and more\textsuperscript{38}.\textsuperscript{5}
Frameshift mutations resulting from insertions and deletions create alternative open reading frames (ORFs) with novel tumor-specific sequences completely distinct from wild-type antigens. A pan-cancer analysis of 19 cancer types from the Cancer Genome Atlas (TCGA) demonstrated that frameshift-derived neoantigens were present in every cancer type. This mutation type also occurs frequently in microsatellite unstable colon cancers (MSI-CRCs) and correlates with higher CD8+ T cell infiltrate in the tumors. For calling indels, in addition to Strelka, EBCall demonstrates the least sensitivity to coverage variability. Pindel specializes in calling larger indels, from 0.50-10 kilobases in length, and structural variants.

Translocations may result in tumor-specific fusion genes, which can alter the reading frame and provide novel junction sequences. Researchers recently investigated the presence of translocations in osteosarcoma, characterized by high genomic instability, and discovered multiple fusion-derived junction-spanning neoantigens. Neoantigen prediction for novel sequences resulting from inter- and intrachromosomal rearrangements in mesothelioma also resulted in multiple neoantigens per patient. Many tools have been developed to predict fusion genes from RNA-seq and/or whole genome sequencing (WGS); recent tools include pizzly, STAR-fusion, JAFFA, GFusion, and INTEGRATE (Table 1.1). The main limitation of these fusion callers is low overlap between tools; they largely achieve high sensitivity at the cost of low specificity. The presence of many false positives makes accurate detection difficult, but this can be mitigated by using multiple tools and requiring predictions to be supported by multiple callers and/or data types (e.g. WGS and RNA-seq).

In addition to mutation-derived neoantigens from known protein coding genes, noncoding regions have immunogenic potential. Noncoding transcripts can be created from noncoding exons, introns, and UTRs, as well as non-canonical reading frames in the coding region.
Laumont et al. investigated traditionally noncoding sequences using liquid chromatography tandem-mass spectrometry (LC-MS/MS) and RNA sequencing (RNA-seq) in leukemia and lung cancer patients and found an abundance of noncoding antigens, both mutated and unmutated.\(^{35}\)

Recent publications have shown that aberrant tumor-specific splicing patterns can create neoantigens. Smart et al. found an approximately 70% increase in total predicted neoantigens after including retained intron sequences along with SNVs in the prediction pipeline.\(^{54}\) Novel junctions created by exon skipping events, or neojunctions, have been shown to create neoantigens.\(^{36}\) Tumor-specific splicing patterns can also cause distinct alternative 3’ or 5’ splice sites, known as splice-site-creating mutations, and these mutations are predicted to create an average of 2.0-2.5 neoantigens per mutation.\(^{55}\)

In addition to the above neoantigen sources, many alternative sources can create neoantigens. For example, V(D)J recombination and somatic hypermutation generate immunoglobulin (Ig) variable region diversity in B and T lymphocytes, and the resulting unique receptor sequences can function as neoantigens in heme malignancies.\(^{37,56}\) Further, researchers have demonstrated that peptides with post translational modifications (PTM), including phosphorylation and O-GlcNAcylation, in primary leukemia samples can serve as MHC-I restricted neoantigens.\(^{57,58}\) Alternative translation events resulting from non-AUG start codons and viral sequences associated with tumors (e.g. HPV) are also a source of neoantigens.\(^{59-63}\) Overall, neoantigen identification requires a sensitive, accurate and comprehensive somatic variant calling pipeline capable of robustly detecting all variant classes relevant for a tumor type (Table 1.2).

1.4 HLA typing, expression, and mutation analysis

T cell priming depends in part on neoantigen presentation on the surface of dendritic cells (DCs), a type of professional antigen presenting cells (APCs). DCs engulf extracellular proteins,
process the peptides, and present the neoantigens on MHC I or II molecules. MHC in humans is encoded by the HLA gene complex, located on chromosome 6p21.3. This locus is highly polymorphic, with over 12,000 established alleles and more in discovery. Because HLA genes are extensively individualized, precise HLA haplotyping is essential for accurate neoantigen prediction. The gold standard for this process is clinical HLA typing using sequence-specific PCR amplification. More recently, NGS platforms such as Illumina MiSeq and PacBio RSII have been combined with PCR amplification to sequence the HLA locus. However, clinical typing can be laborious and expensive, so a common alternative approach is computational HLA typing using the patient’s WGS, WES, and/or RNA-seq datasets, which is typically created from a peripheral blood sample, except in heme malignancies, where a skin sample is often used (Table 1.2).

HLA class I typing algorithms (Table 1.1) have reached up to 99% prediction accuracy when compared to curated clinical typing results. While many class I typing algorithms exist, Optitype, Polysolver, and PHLAT currently have the highest reported accuracies. Despite high precision for class I tools, class II HLA typing algorithms remain less reliable and require additional development to improve prediction accuracy. Few benchmarking studies that consider class II algorithm accuracy have been performed, but a combined class I and II comparison demonstrated that PHLAT, HLA-VBSeq, and seq2HLA performed well with WES and RNA-seq. Additional HLA typing algorithms, xHLA and HLA-HD, have recently been published and show comparable accuracies to the above tools in the studies’ respective comparison metrics.

Since tumor-specific T cell recognition relies on efficient antigen presentation by tumor cells, one mechanism of resistance to immunotherapies is loss or attenuated expression of the
HLA gene loci. Recently, researchers have identified transcriptional HLA repression in a patient with merkel cell carcinoma (MCC) following treatment with autologous T cell therapy and ICB \textsuperscript{76}. The authors found that the transcriptional silencing is reversible in \textit{ex vivo} cultures via treatment with 5-aza and other hypomethylating agents, indicating that reversing epigenetic silencing of the HLA genes could sensitize tumors exhibiting HLA downregulation to immunotherapies \textsuperscript{77}.

Genetic changes at the HLA locus can be determined by Polysolver \textsuperscript{70}, an algorithm that detects HLA-specific somatic mutations from computational HLA typing and variant calling of the tumor HLA locus. Somatic mutation analysis of HNSCC, lung cancer, and gastric adenocarcinoma cohorts demonstrated that HLA mutations are prevalent in all three cancer types \textsuperscript{78–80}. In addition, HLA mutations are enriched towards the beginning of the genes (particularly frameshifts, nonsense and splicing mutations) or within functional domains, where they would be expected to result in a loss of function (LOF) phenotype \textsuperscript{70}. Another tool, LOHHLA, can identify copy number variations in the HLA locus that result in loss of heterozygosity \textsuperscript{81}.

Additional components of the antigen presenting machinery, including B2M and TAP, have been shown to accrue mutations and exhibit altered expression patterns in tumors. In lung and MSI-unstable colorectal cancer, mutations or biallelic loss of \textit{B2M} causes lack of class I HLA presentation \textsuperscript{82,83}. Downregulation of \textit{B2M}, \textit{TAP1}, and \textit{TAP2} expression have also been shown to inhibit tumor antigen presentation \textsuperscript{84,85} and correlate with metastatic breast cancer phenotypes \textsuperscript{86}. Identifying and characterizing altered HLA and associated presentation genes will allow clinicians to prioritize neoantigens that bind to expressed and unmutated alleles.
1.5 Predicting peptide processing

Recognition of a pMHC complex by the T cell is a complex process with many steps and requirements. Most of the attention in the field has been focused on predicting the binding affinity between the patient’s MHC molecule and a given peptide sequence, as this is believed to provide much of the specificity of the overall recognition \(^{87}\). However, even if a peptide has strong MHC binding prediction, the prediction may be meaningless if upstream processing prevents actual loading of that peptide. In general, pipelines generate k-mer peptides via a sliding window applied to the mutant protein sequence, which are subsequently fed into algorithms to predict the affinity of the peptide to the corresponding MHC. However, not all k-mers are capable of being generated \textit{in vivo} due to the limitations of the immune proteasome. Additionally, only a subset of generated peptides will be transported into the appropriate cellular compartments and interact with MHC molecules. These aspects of peptide processing, specifically immune proteasome processing and peptide cleavage, must be considered and several tools have been developed to address this component specifically \(^{88}\).

For both the MHC class I and II pathways, an important upstream step prior to pMHC interaction is proteolysis, which refers to the process of degradation of proteins into peptides, particularly by the immunoproteasome. Multiple tools are now available to capture the specificity of proteasomes and predict the cleavage sites that different proteases target, including NetChop20S \(^{89}\), NetChopCterm \(^{89}\) and ProteaSMM \(^{89,90}\) for MHC class I antigens, and more recently, PepCleaveCD4 and MHC NP II for MHC class II antigens \(^{91,92}\). Algorithms developed in this area are generally trained on two different types of data, \textit{in vitro} proteasome digestion data or \textit{in vivo} MHC-I and II ligand elution data. The neural network-based prediction method NetChop-3.0 Cterm has been shown to perform best at predicting \textit{in vivo} proteolysis for peptide
sources for MHC class I antigen presentation. Cleavage site predictions for MHC class II epitopes show promise, but have yet to be validated for predicting immunogenicity.

For MHC class I antigen processing, peptide fragments are generated from proteins present in the cytoplasm and transported by the transporter associated with antigen processing (TAP) protein into the endoplasmic reticulum (ER), where the peptide is loaded onto an MHC molecule. Thus, in addition to tools focusing on the process of proteolysis, other tools have also been developed for predicting efficiency of peptide transportation based on affinity to TAP proteins. Different methods have been employed in an attempt to classify which peptides have high affinity for TAP binding, including simple/cascade support vector machine (SVM) models and weight matrix models. To address the entirety of this process, the Immune Epitope Database (IEDB) has also developed a predictor for the combination of these processes (proteasomal cleavage/TAP transport/MHC class I).

For the MHC class II pathway, peptides are mostly exogenous and enter the endosome of APCs through endocytosis. As endosomes mature into late endosomal compartments, acidity levels increase and serine, aspartic and cysteine proteases are activated. Proteins, exposed to a series of proteases, are then degraded into potential antigens for presentation. MHC class II molecules are assembled in the ER and transported to these high acidity late endosomes, also known as MHC-II compartments (MIIC). Here, peptides can bind to class II molecules and are protected from destructive processing. In contrast to protein denaturation in the MHC class I processing pathway, cleavage in the MHC class II pathway occurs on folded proteins. Predictors for class II peptide preprocessing prior to MHC binding show the important role that secondary structures play in such reactions, since multiple measures related to secondary structures were found to be highly correlated with the predicted cleavage score. Consideration of secondary
structure will be critical to the future development of tools predicting class II processed peptides. However, while the class I antigen processing pathway has been studied extensively, researchers have only recently started focusing on class II-specific neoantigens as promising results have been shown in cancer immunotherapies. There remains a great need to develop supporting tools and algorithms for class II specific neoantigens.

For the purposes of neoantigen prioritization, it is important to take into account processing steps such as peptide cleavage and TAP transport when using binding prediction algorithms that were trained on in vitro binding data. Recently published binding prediction algorithms have been transitioning to training on data generated in vivo, where processing steps are accounted for intrinsically.

1.6 MHC binding prediction

Neoantigen characterization pipelines have been established specifically to predict the binding of neoantigens to the patient’s specific class I and II MHC molecules (based on HLA typing). Algorithmic development and refinement of reference data sets is very active in this area. We will describe the current state of the art with respect to algorithmic innovation and refinement of the major classes of data (largely in vitro binding assays involving specific MHCs and peptide libraries or MS based approaches) used to train these algorithms.

Peptides bind MHC molecules at a membrane-distal groove formed by two antiparallel α-helices overlaying an eight-strand β-sheet. The peptide-binding region (PBR) of the MHC protein is encoded by exons 2 and 3 of the corresponding HLA gene. High allelic polymorphism allows the binding pocket of MHC molecules to recognize a range of different peptides sequences, particularly the positions involved in anchoring of the peptide to the MHC molecule which vary for each HLA allele. The algorithms and training datasets for predicting
pMHC binding remain an active area of development. Various methods have been employed in an attempt to capture characteristics of peptide and MHC molecules with high probability of binding (Table 1.1).

Early algorithms have mostly focused on training using in vitro pMHC binding affinity measurement datasets. While MHC peptide binding is thought to be the most selective step throughout the antigen presentation process, sole consideration of these predictions still results in high rates of false positive predictions of neoantigens for applications in personalized immunotherapy. This is likely due to the influence of other factors including preprocessing of peptides, stability of the pMHC complex, and peptide immunogenicity. Recently published MHC binding algorithms either use only peptidome data, generated from in vivo immunoprecipitation of pMHC complexes followed by MS characterization, or an integration of MS and binding affinity data. By directly examining ligands eluted from pMHC complexes identified in vivo, predictive models can capture features unique to peptides that have undergone the entire processing pathway. Over 150 HLA alleles have corresponding binding affinity datasets available in IEDB (with highly variable amounts of data for each allele). In contrast, MS peptidome datasets are available for approximately 55 HLA alleles, likely due to lack of high-throughput characterization assays. However, continuous development in MS profiling techniques may soon close the gap between the two types of data. Zhao and Sher recently performed systematic benchmarking for 12 of the most popular pMHC class I binding predictors with NetMHCpan4 and MHCflurry determined to be highest in accuracy for binding/non-binding classifications. The analysis also revealed that incorporation of peptide elution data from MS experiments has indeed improved the accuracy of recent predictors when evaluated using high quality naturally presented peptides.
Different types of algorithmic approaches have been used to model and make predictions for MHC class I molecules. Initially, predictors relied on linear regression (LR) algorithms and more specifically Stabilized Matrix Methods, such as SMM\textsuperscript{111}, SMMPMBEC\textsuperscript{112} and Pickpocket\textsuperscript{113}. However, recently published or updated predictors almost exclusively employ variations of neural networks\textsuperscript{87,102,104,114} as shown in Table 1.3. Linear regression assumes a linear contribution of individual residues to the overall binding affinity; however, while artificial neural networks require more training data, they are able to capture the nonlinear relationship between the peptide sequence and the binding affinity for corresponding MHC molecules through hidden layers in its network architecture. Given the growing number of available training datasets, applications of artificial neural networks have been able to achieve higher accuracy, compared to LR predictive methods\textsuperscript{110}.

While prediction algorithms for MHC class I molecules are well developed, algorithms for MHC class II are fewer, less recently developed, and trained with smaller datasets. Unlike MHC class I molecules, class II molecules are heterodimeric glycoproteins including an α-chain and β-chain, thus there is increased variability in MHC molecules due to dimerization of highly polymorphic alpha and beta chains. The binding pocket for class II molecules is open on both ends which allows a larger range of peptides to bind. The most frequently observed peptide lengths for class II binding fall between 13 and 25 amino acids (AA)\textsuperscript{115} while those for class I typically fall between 8 to 15 AA\textsuperscript{87}. Though for any one particular MHC allele, the preferred length may be much more constrained to one or two lengths. Algorithms built for class II predictions generally rely on matrix-based methods and ensembles of artificial networks. A selection of popular MHC class II binding prediction algorithms are summarized in Table 1.1\textsuperscript{116}. 
While there is an extensive list of MHC binding prediction tools for both class I and class II molecules, there remains a need to not only expand the training data for a larger range of HLA alleles but also to refine the type of training data being used in these algorithms. Although in vivo MS data will capture features of peptides naturally presented by MHC molecules, whether such peptides are able to induce an immune response remains unknown. Algorithms should ideally incorporate experimentally and clinically validated immunogenic peptides in their training and validation datasets. As ongoing neoantigen clinical trials produce more of such data, tool development and refinement in this area will also be made possible.

1.7 Neoantigen prioritization and vaccine design pipelines

Due to the numerous factors that are involved in the process of antigen generation, processing, binding, and recognition, a number of bioinformatic pipelines have emerged with the goal of assembling available tools to streamline the neoantigen identification process for different clinical purposes (e.g. predicting response to checkpoint blockade therapy, designing peptide or vector based vaccines, etc.). Table 1.1 includes a selection of these pipelines and Table 1.2 provides extensive practical guidance for their use in clinical studies. These pipelines address multiple factors that should be given careful consideration when attempting to predict neoantigens for effective cancer treatments. These considerations include: use of multiple binding prediction algorithms (binding prediction variability/consensus), integration of both DNA and RNA data (neoantigen candidate genes/transcript expression and variant allele expression), variant phasing (proximal variants detected on the same allele will influence neoantigen sequences)\textsuperscript{32,117}, interpretation of variants in the context of clonality/heterogeneity\textsuperscript{118}, patient tumor HLA expression and somatic mutation, and prediction of tumor immunogenicity\textsuperscript{119,120}. These pipelines are then able to provide a comprehensive summary of
critical information for each neoantigen prediction, including: variant identity (genomic coordinates, ClinGen allele registry ID, and HGVS variant name), predicted consequence of the variant on the amino acid sequence, corresponding gene and transcript identifiers, peptide sequence, position of variant within the candidate neoantigen peptide, binding affinity predictions for mutant and the corresponding wild type peptide sequences, agretopicity value (mutant versus wildtype peptide binding affinity)\textsuperscript{121}, DNA variant allele frequency (VAF), RNA VAF and gene expression values for the gene harboring the variant. Additional data on whether peptides are generated from oncogenic genes, peptide stability, peptide processing/cleavage, and peptide manufacturability should also be considered for final assessment of neoantigens (Table\textsuperscript{1.2}).

Several pipelines attempt to integrate DNA and RNA sequencing data by evaluating the VAFs and gene/transcript expression values of mutations. Most pipelines currently take into account SNVs and indels, with only a subset considering gene fusion events\textsuperscript{8,32,122}. Consistent use of the same build/assembly of the genome throughout analysis pipelines, as well as emphasis on quality control (QC) when performing variant detection and expression analysis, are important for ensuring high confidence in variants detected (Table\textsuperscript{1.2}). Once these mutations are confirmed to exist and be expressed, pipelines then generate a list of neoantigen candidates and consider the probability of cleavage, the location of cleavage, and the TAP transport efficiency of each candidate\textsuperscript{8,123,124}. Binding affinities of peptides to the patient-specific MHC molecules are subsequently predicted by use of one or more algorithms (Table\textsuperscript{1.1}). However, binding affinity predictions by multiple prediction algorithms vary and best practices for determining a consensus are poorly articulated at this time. Furthermore, the gold standard independent validation datasets that exist to evaluate the accuracy of divergent predictions are
limited. It remains to be validated whether combining multiple prediction algorithms increases the true positive rate of neoantigen predictions. Some pipelines also consider manufacturability by measuring peptide characteristics, immunogenicity by comparison of either self-antigens defined by the reference/wildtype proteome or known epitopes from viruses and bacteria provided by IEDB, and pMHC stability.

Pipelines vary in their choices of how to rank neoantigens and which specific type of algorithm to use when performing such calculations. Thus, a major challenge lies in how each component should be weighted to create an overall ranking of neoantigens in terms of their potential effectiveness. Kim et al. have attempted to capture the contributions of nine immunogenicity features through training of machine-learning based classifiers. However, high-quality and experimentally-validated neoantigens for training such models remain extremely sparse. That is, there is no consensus on the features of a ‘good’ neoantigen, capable of inducing T cell responses in patients. Furthermore, clinicians may need to consider customized filtering and ranking criteria for individual patient cases, tumor types, or clinical trial designs, details that are not well supported by existing pipelines. For these reasons, clinical trial efforts should establish an interdisciplinary team of experts analogous to a molecular tumor board for formal quantitative and qualitative review of each patient’s neoantigens. While pipelines such as pVACtools and Vaxrank are designed to support such groups, there are many important areas that current pipelines can improve upon, including 1) consideration of whether the mutation is located at anchor residues for each HLA allele, 2) somatic mutation and expression of patient specific HLA alleles, 3) expression level of important cofactors such as genes involved in processing, binding and presentation, and 4) additional factors that influence manufacturing and delivery of predicted neoantigens.
1.8 Peptide creation, delivery mechanisms and related analysis considerations for vaccine design

Once neoantigen prioritization is complete, personalized vaccines are designed from predicted immunogenic candidate sequences. Multiple delivery mechanisms exist for use in clinical trials; these include synthetic peptides, DNA, mRNA, viral vectors, and *ex vivo*-loaded dendritic cell vaccines\(^{126,127}\). Cancer vaccine delivery is an extensive topic out of the scope of this review; however, some existing reviews discuss this topic in detail\(^ {126-128}\). Once a mechanism is chosen and the vaccine is delivered to the patient, professional APCs endocytose the neoantigen sequences. Then, they are processed to generate class I and II-restricted MHC peptides for presentation and T cell activation. To design a successful delivery vector, additional analysis steps are necessary to assess peptide manufacturability and avoid potential incidental DNA vector junctional epitope sequences, or junctions spanning neoantigen sequences that create unintended immunogenic epitopes\(^ {8,129}\).

Synthetic long peptides (SLPs) are an effective neoantigen delivery mechanism in personalized immunotherapy preclinical studies and clinical trials\(^ {30,101,130,131}\). These peptides are created from sequences of 15-30 amino acids that contain a core predicted neoantigen. SLPs have greater efficacy than short synthetic peptides, 8-11 amino acids, because longer peptides require internalization and processing by professional APCs, while short peptides can induce immunological tolerance by binding directly to MHC-I on non-professional APCs\(^ {132-134}\). One limitation of SLPs is manufacturability. Certain chemical properties of the amino acid sequence can make peptides difficult to synthesize and longer peptides can encounter solubility problems (i.e. they become insoluble). Vaxrank\(^9\) aims to address these concerns by incorporating a manufacturability prediction step in the neoantigen prioritization pipeline. This step measures
nine properties that contribute to manufacturing difficulty, including hydrophobic sequences, cysteine residues, and asparagine-proline bonds. The algorithm then uses this information to choose an ideal window surrounding the somatic mutation for optimum synthesis.

DNA vectors have also successfully delivered neoantigens in a recent preclinical study\textsuperscript{135}, and DNA neoantigen vaccine clinical trials are currently ongoing in pancreatic and triple negative breast cancer\textsuperscript{136}. Neoantigen encoding DNA sequences can be either directly injected via plasmid vectors using electroporation or incorporated into viral vectors for delivery to patient cells. Adenovirus and vaccinia are the most common viral vectors for personalized vaccines; both are dsDNA viruses that can incorporate foreign DNA\textsuperscript{137}. To maximize neoantigen effectiveness for both vectors, researchers must design sequences with effective junctions and/or spacers. This ensures correct cleavage of the combined sequence by the proteasome as well as the avoidance of inadvertent immunogenic junction antigens. Multiple methods exist to address these challenges. Furin is a peptidase in the trans-Golgi network that cleaves immature proteins at sequence-specific motifs\textsuperscript{138}. Recently, furin-sensitive cleavage sequences were incorporated into a neoantigen DNA vaccine to cleave the sequence into functional neoantigens\textsuperscript{135}. EpiToolKit\textsuperscript{123} addresses incorrect peptide cleavage in its pipeline by incorporating NetChop\textsuperscript{89}. This tool predicts proteasomal cleavage sites for each neoantigen to exclude candidates that would undergo inappropriate cleavage. pVACvector, an algorithm included in pVACtools\textsuperscript{8}, optimizes neoantigen sequence order by running pVACseq on the junction sequences and prioritizing those with low immunogenicity. If high junction immunogenicity cannot be avoided, spacer sequences are included to decrease the potential for inadvertent neoantigens. Taking such analytical considerations into account for personalized vaccine design ensures maximum treatment efficacy in patients.
1.9 T cell recognition, TCR profiling, and immune cell profiling to evaluate response

The ultimate objective of introducing a neoantigen-derived vaccine is to elicit and/or expand a tumor-specific T cell response. This can be evaluated by experimental methods that measure T cell activation and activity, or computational methods that characterize the patients TCR repertoire prior to and after immunotherapy. Standard methods such as IFN-gamma ELISPOT assays\textsuperscript{139} or MHC multimer assays\textsuperscript{140} are out of the scope of this review, but have been used widely for neoantigen validation purposes\textsuperscript{28,141}. T cells individually undergo complex combinatorial rearrangements in the T cell receptor gene loci in order to create unique clonotypes responsible for recognizing antigens. This process occurs within the V(D)J region of the gene, particularly the complementarity-determining region 3 (CDR3), encoding a region of the TCR important for recognizing the pMHC complex. Thus, attempts to characterize the TCR repertoire focus on the identification and characterization of CDR3 sequences, representative of the unique T cell clones. This process, termed TCR clonotyping, has been used to identify clonal T cell responses to neoantigens following vaccination with a personalized cancer vaccine or checkpoint blockade therapy\textsuperscript{28}. Researchers have also established an association between the size and diversity of a patient’s TCR repertoire and their response to cancer immunotherapies\textsuperscript{142}. While changes in the clonality and diversity of the TCR repertoire, observed from either peripheral blood or tumor-infiltrating lymphocytes (TIL), suggest that an anti-tumor T cell response is occurring, they are global metrics that do not successfully identify the T cell clonotypes responsible for tumor rejection.

There are a variety of technologies and tools available that allow sequencing and subsequent analysis of the TCR repertoire. Commercial services such as Adaptive, ClonTech and iRepertoire differ in a number of aspects including the required starting material, library preparation
methods, the targeted TCR chains and/or CDR regions for sequencing, supported organisms, as well as sequencing platforms used\textsuperscript{143}. Several tools exist to identify TCR CDR3 sequences using various types of data, such as output data from focused assays (e.g. Adaptive, ClonTech, CapTCR), bulk tumor RNA-seq\textsuperscript{144} and single cell RNA-seq\textsuperscript{144,145}, particularly from the TCR alpha and beta genes (\textit{TRA}, \textit{TRB}). Challenges associated with TCR profiling include the diversity of the repertoire itself, correctly determining the pairing of \textit{TRA} and \textit{TRB} clonotypes, and the subsequent analysis or validation necessary to pair T cell clones with their target neoantigens. Studies have quantified or predicted the T cell richness, or total number of T cell clones, in the peripheral blood of a healthy individual up to $10^{19}$\textsuperscript{146}. Thus, there is a sampling bias, based upon the blood draw that was taken, the sample used for sequencing, and the input material for library preparation, that prevents complete evaluation of the global T cell repertoire.

TCR profiling requires alignment of sequencing reads to the reference TCR genes and assembly of the rearranged clonotypes. MixCR has been used for TCR alignment and assembly in both bulk and single cell methods\textsuperscript{144,147}. MIGEC\textsuperscript{148} is utilized for methods involving the use of unique molecular identifiers, while TraCeR is designed specifically for single cell methods\textsuperscript{145}. MiXCR recovers TCR sequences from raw data through alignment and subsequent clustering, which allows grouping of identical sequences into clonotypes. If sequences are generated from bulk material (e.g. whole blood, bulk TIL), \textit{TRA} and \textit{TRB} sequences cannot be paired to specifically define the T cell clonotypes. They may be inferred based upon frequency; however, due to the very high diversity of the T cell repertoire, there are often many clonotypes at similar or low frequencies that make deconvolution of \textit{TRA/TRB} pairs difficult. With the advent of single cell sequencing data, tools, such as TraCeR, are now able to identify paired alpha/beta
sequences within individual cells that have the same receptor sequences and thus have been derived from the same clonally expanded cells \textsuperscript{145}.

Identification of clonally expanded neoantigen-specific TCRs complements neoantigen prediction and characterization by indicating whether an active T cell response has been stimulated by an immunotherapeutic intervention. Lu et al. recently developed a single cell RNA-seq approach that identifies neoantigen specific TCRs by culturing TILs with tandem minigene (TMG)-transfected or peptide-pulsed autologous APCs \textsuperscript{149}. Experimental validation data of individual neoantigens can then be utilized for training and improvement of current neoantigen prioritization strategies.

The clonality of the TCR repertoire can be further evaluated to identify T cell clones that may recognize the same neoantigen. Studies have identified oligoclonal T cell populations that converge, with consistent CDR3 motif sequences, to recognize the same neoantigen \textsuperscript{150}. Taking into account the diversity of the repertoire, these findings suggest that oligoclonal events are more likely than monoclonal events, and there is less likely to be a one-to-one mapping between T cell clones and neoantigens. Oligoclonal events and the convergence of the T cell repertoire can be better studied with tools such as GLIPH, which was developed to identify consistent CDR3 motifs across \textsuperscript{151} T cells in bulk TCR sequencing.

Anti-tumor T cell responses have been correlated with changes in the infiltrating immune microenvironment. Methods such as CIBERSORT have been developed for characterizing cell compositions based on gene expression profiles from tumor samples \textsuperscript{152}. Association between immune cell infiltrates and various factors including somatic mutation, copy number variation and gene expression can be explored interactively through TIMER \textsuperscript{153}. This topic has been reviewed in more depth elsewhere \textsuperscript{154}. A larger selection of available tools related to T cell and
immune cell profiling are listed in Table 1.1. Overall, few studies have focused on the integration of T cell profiling with neoantigen detection, with the exception of Li, et al. 2016, where identification of TCR clones from RNAseq samples across TCGA samples was compared to the mutational profiles of tumors, successfully identifying several public neoantigens shared across individuals. Due to the availability of peripheral blood samples and TCR sequencing data with matched tumor DNA/RNA sequencing, one major development in the field remains the aggregation of this data and the appropriate supervised approach to identify TCR/neoantigen pairs, leveraging the data available to enhance identification of both neoantigens and optimize personalized medicine approaches to cancer immunotherapy.

1.10 Conclusions and future directions

While great strides have been made in developing pipelines for neoantigen identification, there is significant room for improvement. Tools for the rational integration of the myriad complex factors described above are needed. In some cases, useful tools exist but have not been incorporated into analysis workflows. In other cases, factors we believe are important are not being considered due to a lack of tools.

Variant types beyond SNVs and indels have been confirmed as neoantigen sources, but there remains little support for them in current pipelines. Fusions have recently been incorporated into pipelines such as pVACfuse, a tool within pVACtools, INTEGRATE-neo, and NeoepitopePred. However, additional genomic variant types that lead to alternative isoforms and expression of normally non-coding genomic regions remain unsupported, in spite of preliminary analyses suggesting their importance. An additional orthogonal, but poorly supported neoantigen source, is the proteasome, which was found to be capable of creating novel antigens by splicing peptides from diverse proteins to create a single antigen. Several tools
exist to computationally predict PTMs and alternative translation events from sequencing data, such as GPS\textsuperscript{157} and KinasePhos\textsuperscript{158} for phosphorylation events and altORFev\textsuperscript{159} for alternative ORFs. To determine the immunogenicity of these alternate peptides, any tumor-specific predicted sequences could be input into neoantigen prediction software.

Low accuracy with class II HLA typing algorithms has impeded extensive class II neoantigen prediction. When clinical class II HLA typing data is available, it should be used in place of computational HLA predictions in pipelines to improve prediction reliability. In addition, while somatic alterations in \textit{HLA} gene loci and antigen presentation machinery have been implicated in immunotherapeutic resistance, these properties have not been leveraged in predicting neoantigen candidates. The \textit{HLA} gene expression is more often summarized at the gene, not the allele level. Furthermore, HLA expression is commonly determined from bulk tumor RNAseq data, which contains normal, stromal, and infiltrating immune cells, that may all express these genes. The relationship between the present HLA alleles and predicted neoantigen profile has not been studied, and it remains to be seen whether neoantigens restricted to absent or mutant HLA alleles should be specifically filtered out.

For the neoantigen prediction step, mutation positions in the neoantigen should be carefully considered if occurring in anchor residues since this results in a core sequence identical to the wild type protein. There is also a bias towards class I neoantigen prediction because there is less binding affinity training data and fewer algorithms for class II due to increased MHC binding complexity. Studies have also shown low consensus across MHC binding predictors\textsuperscript{8}. pVACtools\textsuperscript{8} addresses this challenge by running multiple algorithms simultaneously and reporting the lowest or median score, but a more definitive method for binding affinity consensus remains to be developed. Neoantigen prediction pipelines could also benefit from the inclusion
of information on the proposed delivery mechanism to improve prioritization and streamline vaccine creation.

Although TCR sequences have been recognized to be highly polymorphic, TCRs from T cells recognizing the same pMHC epitope may share conserved sequence features. Researchers have started to quantify these predictive features with the hope of modeling epitope-TCR specificity. Multiple tools (e.g. TCRex, NetTCR, Repitope) now attempt to predict TCR-epitope binding when given specific TCR sequences. By taking into account the binding specificity of the patient’s existing TCR sequences, neoantigen candidates can be further prioritized according to immunogenicity. A major advance to optimize treatment strategies may require integration of pipelines that perform all of the steps necessary for the generation and processing of neoantigens and the identification of T cell clones that efficiently recognize them.

Implementing a set of best practices to predict high quality immunogenic neoantigens can lead to improved personalized patient care in the clinic. However, predicting and prioritizing neoantigens is a complicated process that involves many computational steps, each with individualized, adjustable parameters (we provide a specific end to end workflow based on our current practices at https://pmbio.org/). Given this complexity, review of candidates by an immunogenomics tumor board with diverse expertise is highly recommended. We have outlined each step in the neoantigen workflow with human clinical trials in mind, but further research is needed in model organisms to facilitate development of immunotherapies for human use. Improving neoantigen characterization tools to support in silico modeling of immune response, model organism systems, human derived samples, and human patient trials is an essential step for improving patient response rates across cancer types.
1.11 Acknowledgements

We are grateful to the research and clinical trial participants and their families, without whom none of this would be possible. We would like to thank Christopher Miller, Jasreet Hundal, Susanna Kiwala, Jason Walker, Thomas Mooney, Adam Coffman, Cody Ramirez, Zachary Skidmore, Michael McLellan, Michelle Becker-Hapak, Simon Goedegebuure, Tammi Vickery, Tanner Johanns, Gavin Dunn, Todd Fehniger, Antoni Ribas, Elaine Mardis, and Robert Schreiber for invaluable discussions on relevant immunology, immunogenomics, bioinformatics and analysis concepts. We also thank Joshua McMichael for guidance on the creation of Fig. 1.1. OLG was supported by the NIH National Cancer Institute (U01CA209936, U01CA231844, and U24CA237719). MG was supported by the NIH National Human Genome Research Institute (R00HG007940), the NIH National Cancer Institute (U01CA209936, U24CA237719), and the V Foundation for Cancer Research.

1.12 Author contributions

MMR, HX, KMC, OLG, and MG wrote the manuscript and prepared the figures and tables with input from WEG. All authors reviewed and approved the final version of the manuscript.
Chapter 2: Computational prediction of MHC anchor locations guides neoantigen identification and prioritization

2.1 Preamble

The following chapter has been published:


As an author of the published manuscript, and in compliance with the editorial policies at Science Immunology, the cited publication is included in full in the following chapter.

2.2 Introduction

Neoantigens arise from short peptide sequences specifically found in tumor cell populations resulting from sources such as somatic mutations, RNA editing\textsuperscript{161}, alternative splicing\textsuperscript{162}, etc. They can be loaded onto major histocompatibility complex (MHC) class I or II molecules to allow recognition by cytotoxic or helper T cells. Upon recognition, T cells are then able to signal cell death for an anti-tumor response. Multiple studies have shown the potential of neoantigen based immunotherapy treatments for cancer\textsuperscript{4,30,131} and numerous clinical trials are underway. Accurate neoantigen prediction and prioritization is critical to understanding tumor immunology,
response to checkpoint blockade therapy, and for the design of personalized vaccines and T cell therapies. Several bioinformatic tools and pipelines have been developed to facilitate neoantigen identification.

The effectiveness of a neoantigen-based vaccine relies in part on whether the neoantigen sequences presented to T cells have previously been exposed to the immune system and would be subject to central tolerance (where immune response to antigens is limited as a result of clonal deletion of autoreactive B cells and T cells). While a variety of mutation types are being explored as neoantigen sources, the vast majority of somatic mutations currently identified as sources of neoantigens are single nucleotide variants (SNVs), though emerging research suggests that we are underestimating the potential for neoantigen generation from more complex forms of variation. Amino acid sequence changes between the wildtype (WT) and mutant (MT) peptides are subtle (often a single amino acid substitution) and mutant peptides remain similar to native sequences of the host. Additionally, only a subset of positions on the loaded peptide sequence are primarily presented to the T-cell receptor for recognition, and another subset of positions are primarily responsible for anchoring to the MHC, making these positional considerations critical for predicting T-cell responses. Thus, subtle amino acid changes must be interpreted cautiously. Multiple factors should be considered when prioritizing neoantigens, including mutation location, anchor position, predicted MT and WT binding affinities, and WT/MT fold change, also known as agretopicity. Examples of the four distinct possible scenarios for a predicted strong MHC binding peptide involving these factors are illustrated in Figure 2.1. There are other possible scenarios where the MT is a poor binder, however those are not listed as they would not pertain to our goal of neoantigen identification. The first scenario shows the cases where the WT is a poor binder and the MT peptide, a strong
Figure 2.1. Anchor and mutation position scenarios at the MHC-peptide-TCR interface.

Illustration of MHC-peptide-TCR interface using an example structure with anchors at position 2, 5 and 9. At the contact interface between the peptide-loaded MHC and the recognizing T-cell receptor, certain positions are responsible for anchoring the peptide to the MHC molecule and/or potentially being recognized by the TCR. The position of tumor specific (“mutant”) amino acids relative to anchor positions, and predicted binding affinity of mutant and wild type peptides produce four distinct scenarios for interpreting candidate neoantigens. Example TCR recognition sites are shown in blue while MHC anchor locations are shown in green. The peptide residues are shown in orange while the mutant residue is marked with red. A yellow force field with varying density is used to illustrate binding strength between peptide and the MHC molecule. Three different cases of TCR recognition level are depicted including: self-recognizing TCR absent due to negative selection, weak-recognizing TCR due to weak MHC binding of presented peptide, and strong-recogni

binder, contains a mutation at an anchor location. Here, the mutation results in a tighter binding of the MHC and allows for better presentation and potential for recognition by the TCR. As the WT does not bind (or is a poor binder), this neoantigen remains a good candidate since the sequence presented to the TCR is novel. The second and third scenarios both have strong binding WT and MT peptides. In the second scenario, the mutation of the peptide is located at a non-
anchor location, creating a difference in the sequence participating in TCR recognition compared
to the WT sequence. In this case, although the WT is a strong binder, the neoantigen remains a
good candidate that should not be subject to central tolerance. However, as shown in the third
scenario, there are neoantigen candidates where the mutation is located at the anchor position
and both peptides are strong binders. Although anchor positions can themselves influence TCR
recognition\textsuperscript{173}, a mutation at a strong anchor location generally implies that both WT and MT
peptides will present the same residues for TCR recognition. As the WT peptide is a strong
binder, the MT neoantigen, while also a strong binder, will likely be subject to central tolerance
and should not be considered for prioritization. The last scenario is similar to the first scenario
where the WT is a poor binder. However, in this case, the mutation is located at a non-anchor
position, likely resulting in a different set of residues presented to the TCR and thus making the
neoantigen a good candidate. Recent studies on neoantigens for both mouse and human models
have confirmed the importance of anchor location when predicting the overall immunogenicity
of a given peptide\textsuperscript{174,175}. However, it is important to note that these scenarios are not absolute.
While a position that acts as a strong anchor is less likely to be TCR facing (and vice versa),
certain positions of the peptide may interact to varying degrees with both the MHC and TCR.

The mutation’s position within the peptide relative to its anchor positions for the patient’s
human leukocyte antigen (HLA) alleles is currently overlooked by neoantigen prediction
pipelines. However, failing to account for these positional considerations may result in
susceptibility to central tolerance and potentially induce auto-immunity. Many recently
published neoantigen studies have used simple filtering strategies with either only binding
affinity filters\textsuperscript{131,176} (e.g. MT peptide IC50 < 500 nM) or with an additional agretopicity filter\textsuperscript{177--}
179, all without specifying whether they account for anchor and mutation locations during their
selection process. Researchers have previously discussed how anchor locations can affect our interpretation of other factors considered in neoantigen prioritization (e.g. MT, WT binding affinities). However, a systematic method for determining anchor locations for the wide range of HLA alleles present in the population and application of these to evaluate MT/WT peptide pairs arising in tumors has not been reported. As a result, many neoantigen studies have either failed to adequately consider this crucial factor or have used simplified and arbitrary assumptions to guide their neoantigen identification process.

Here, we provide a computational workflow for predicting anchor locations for a wide range of HLA alleles using a seed dataset generated from a collection of patient samples from local tumor sequences studies combined with samples from The Cancer Genome Atlas (TCGA). Analysis of results showed clusters of different anchor trends among the HLA alleles analyzed and a subset of these HLA anchor results were orthogonally validated using protein crystallography structures. Representative anchor trends were experimentally validated using cell-based stability assays and IC50 binding assays. Using additional TCGA samples, we further evaluated how prioritization results may change when provided with additional anchor information, emulating a subset of steps in the neoantigen selection process by an immunotherapy tumor board, tasked with prioritizing vaccine candidates. By sharing our results for incorporation into neoantigen prediction pipelines, we hope to improve neoantigen prioritization for relevant experimental and clinical studies.
2.3 Results

2.3.1 Computational and quantitative prediction of HLA-specific anchor positions

In order to predict anchor locations for a wide range of HLA alleles, we assembled a seed HLA-peptide dataset of strong binding peptides with a median predicted IC50 of less than 500 nM across (up to) 8 MHC class I algorithms (NetMHC\textsuperscript{114}, NetMHCpan\textsuperscript{182}, MHCnuggets\textsuperscript{183}, MHCflurry\textsuperscript{102}, SMM\textsuperscript{111}, Pickpocket\textsuperscript{113}, SMMPMBC\textsuperscript{112}, NetMHCcons\textsuperscript{184}). These peptides were obtained from TCGA and supplemented with additional patient datasets from our own neoantigen study cohorts including lymphoma, glioblastoma\textsuperscript{181}, breast cancer, and melanoma. A total of 609,807 peptides were identified with the majority being 9-mers and 10-mers (Figure 2.2). In total, these peptides corresponded to 328 HLA alleles across 1,443 tumor samples.
Figure 2.2. Distribution of peptides collected per allele across 328 HLA alleles, split by peptide length. Histograms summarizing the distribution of predicted strong binding peptides collected for 328 HLA alleles. Peptides are plotted according to their respective lengths. The total number of peptides for each k-mer is shown as N. The x-axis represents the total number (X) of k-mer peptides for individual HLA alleles while the y-axis represents the number of HLA alleles with X number of peptides matched.

For each individual HLA allele of which data was obtained, peptides were sorted and separated by their respective lengths, ranging from 8 to 11 (Figure 2.3). These peptides were in silico at all possible positions to all possible amino acids. Predicted binding then mutated affinities for each individual peptide were then obtained using the same set of algorithms as previously described. These binding affinities were compared to the median binding affinity of the original strong binding peptide sequence. This comparison enables us to evaluate how mutations occurring at each individual position change the predicted binding interaction between the strong binding peptide and the MHC molecule. A significant change observed at a particular location indicates a higher probability of the amino acid at the position acting as an anchor. On the other hand, little to no change in binding affinities when a position is mutated would indicate a lower probability of the position acting as an anchor. An overall score per position was obtained by summing across all peptides analyzed for an individual HLA allele (Figure 2.3; Figure 2.4).
Figure 2.3. Overview of computational workflow for anchor position prediction. Schematic of our computational workflow to simulate the effect of mutation position on MHC binding affinities of peptides for individual HLA alleles. HLA and peptide pairings are selected from a reference dataset of putative strong binders. All possible amino acid changes are applied to all possible positions and the impact on binding affinity is assessed. An overall peptide anchor score is calculated for each position for all HLA-peptide length combinations.

Fig. 2.4. Saturation analysis for evaluating subsample size of peptides needed for simulation analysis. Results of saturation analyses performed for 8-11-mer peptides using Spearman correlation are depicted as line plots with colors indicating random iterations. For each k-mer, 5 random rounds were performed with subset sizes of 1, 5, 10, 20, 50, 100, 200, 500 and 1,000. Correlations were calculated between each subset size and the largest subset (N=1,000) with the assumption that 1,000 samples was adequate to represent a ground truth for this purpose.

2.3.2 Prediction results show distinct patterns of HLA anchor locations

We generated anchor prediction scores for the 328 HLA alleles with strong binding peptides in our seed dataset. These HLA alleles include 95 HLA-A alleles (representing 99.2% of the HLA-
A alleles observed in the population according to the Allele Frequency Net Database), 175 HLA-B alleles (representing 97.9% of HLA-B alleles) and 58 HLA-C alleles (representing

Figure 2.5. Hierarchical clustering of anchor prediction scores across all 9-mer peptides.
Anchor prediction scores clustered using hierarchical clustering with average linkage for all 318 HLA alleles for which 9-mer peptide data were collected (out of the 328 HLA alleles, 10 did not
have corresponding 9-mer data). For the heatmap, the x-axis represents the 9 peptide positions and the y-axis represents 318 HLA alleles. Example HLA clusters have been highlighted with various color bands and the score trends for individual HLA alleles are plotted. In the cluster line plots on the left, the x-axis shows the peptide positions while the y-axis corresponds to the anchor score, normalized across all peptide positions. Different annotations have been given to help summarize the trends observed in individual clusters, where numbers represent positions and letters represent its strength as a potential anchor in comparison to other anchors (S: strong, M: moderate, W: weak). The median scores for each cluster are presented with a dashed line.

Figure 2.6. Hierarchical clustering of anchor prediction scores across all 8, 10, and 11-mer peptides assembled. Heatmaps depict anchor prediction scores clustered using hierarchical clustering with average linkage across all 328 HLA alleles for which 8-mer, 10-mer and 11-mer peptide data were collected. For the individual heatmaps, the x-axis represents the k peptide positions and the y-axis represents the HLA alleles for which k-mer peptides were available.

98.5% of the HLA-C alleles\textsuperscript{185} (Methods). Results were separated based on peptide lengths (8-11) and the anchor prediction scores across all HLA alleles were visualized using hierarchical
clustering with average linkage (Figure 2.5; Figure 2.6). We observed different anchor patterns across HLA alleles, varying in both the number of anchor positions as well as the location. These anchor position patterns could be roughly clustered into 6 distinct groups. Additionally, while we utilized peptides generated from cancer patient data, we wanted to see if changing the source of the peptides would affect our predicted anchor patterns (Figure 2.7; Methods). After averaging calculations over 1,000 peptides from each source, we observed high convergence of the anchor

**Figure 2.7. Comparison of anchor pattern across different seed peptide sources using HLA-A*02:01.** Anchor pattern for HLA-A*02:01 calculated from 1,000 peptides for each individual peptide source. We compared results from our seed dataset of peptides (tumor mutation derived) to three qualitatively distinct additional sources of peptides: 1) 9-mer peptides generated from a random 100,000,000 length sequence of amino acids, 2) random 9-mer peptides sampled from the human reference proteome (based on known Ensembl protein sequences) and 3) random 9-mer peptides from a viral proteome (variola). From each source, we identified 1,000 peptides that were predicted to be strong binders (Methods) and mutated each position for each amino acid following our computational workflow. We performed in silico predictions for each of the mutated peptides and calculated anchor probabilities based on all peptides collected. Normalized anchor probabilities are plotted on the y-axis and the peptide positions are listed on the x-axis.
pattern for HLA-A*02:01 across all four different sources (random, reference proteome, viral proteome, and cancer mutations). This demonstrated the possibility of expanding results to a wider span of HLA alleles in the future by utilizing different sources of peptides.

Previously, anchor locations have generally been assumed to be at the second and terminal position of the peptide with equal weighting (with the notable exception of HLA-B*08:01)\(^{186}\). Our 9-mer clustering results confirm that the majority of HLA alleles predicted show positions 2 and 9 as likely anchor locations (95% fall into the 2S-9W, 2W-9S, 2M-9M clusters combined). However, three distinct cluster groups can be further identified within the larger group. The 2S-9W cluster represents HLA alleles with a strong anchor predicted at position 2 and a weak anchor predicted at position 9 (2S-9W; Figure 2.5). The 2W-9S cluster shows those with a strong anchor predicted at position 9 and a weak anchor predicted at position 2 (2W-9S; Figure 2.5). Additionally, we observe a smaller cluster of HLA alleles with moderate anchor predictions for both positions (2M-9M; Figure 2.5) and another cluster with strong anchor predictions for only position 9 (9S; Figure 2.5). We also discovered other patterns differing from the previous anchor assumptions of 2 and 9. In particular, we observed a clustered group of HLA-C alleles that have a moderate anchor at 3 and 9 accompanied by a weaker signal at 2 (2W-3M-9M; Figure 2.5). A smaller group of HLA-B alleles also show an additional anchor at position 5 (2W-3W-5W-9M; Figure 2.5). Our results indicate that a conventional anchor assumption putting equal weights on positions 2 and 9 does not capture the significant heterogeneity in anchor usage between different HLA alleles. These anchor considerations can affect neoantigen prioritization decisions and we hypothesized that HLA allele-specific anchor predictions would allow ranking of neoantigens with greater accuracy.
To analyze the possibility of our clusters arising due to bias from prediction algorithms and their training datasets, we examined the training dataset used by NetMHCpan4.0 for the 328 HLA alleles that we have included (Appendix 2). Additionally, for alleles with limited data (hence predicted through a pan allele strategy), the ability to effectively utilize information from other alleles (for those lacking training data) also depends on how similar the allele of interest is to other alleles that do have robust amounts of data. Thus we also looked at the neighboring HLA alleles that are used in cases where training data was not available and the distance (calculated by NetMHCpan4.0) between each HLA allele and its nearest neighbor. Overall, the majority of HLA alleles (99.7%) had binding predictions from at least 4 different algorithms with 54 HLA alleles being supported by all 8 (Figure 2.8a). The amount of training data for each HLA allele also correlated with how consistent binding predictions were across algorithms (Figure 2.8b). This analysis confirms our expectation that more training data leads to more consistent predictions across algorithms. For the smaller clusters such as 2W-3W-5W-9M (blue), 2W-3W-9M (green) and 9S (gray), we can see that in general they fall along the same level as other clusters when comparing their nearest neighbor distances and how much variation is seen across prediction algorithms (Figure 2.8c,d). Additionally, 2 out of the 3 alleles in the 2W-3W-5W-9M cluster, 4 out of 7 alleles in the 2W-3W-9M cluster and 6 out of 6 alleles in the 9S cluster all have some amount of training data available (Figure 2.8e, Appendix 2). When looking at network graphs showing HLA alleles and their neighbors, those with 2W-3W-5W-9M (blue) and 2W-3W-9M (green) patterns are scattered among different clusters with some alleles even acting as the center node (indicating non-zero amount of training data and thus no need to estimate binding based on similar alleles) (Figure 2.8f,g). Thus, we concluded that the less common anchor clusters do not correlate with lack of training data nor similarity to a limited set
Figure 2.8. Analysis of potential for supporting algorithm bias across 328 HLA alleles.  
Bar plot showcasing the number of HLA alleles being supported by the algorithms used.
maximum of 8 algorithms are available with the majority of the HLA alleles being supported by at least 4 algorithms. The specific breakdown is as following: out of the 328 HLA alleles, 54 HLA alleles are supported by all 8 algorithms, 2 are supported by 7 algorithms, 11 are supported by 6 algorithms, 8 are supported by 5 algorithms, 252 are supported by 4 algorithms, and 1 is supported by 1 algorithm. b, Scatter plot showing distribution of HLA allele anchor pattern clusters with respect to two metrics: Mean value of variances calculated based on up to 8 different algorithmic predictions (x-axis) and distance to nearest HLA allele neighbor according to NetMHCpan (y-axis). If the HLA allele has training data available, then the distance is 0 since the nearest neighbor is only used in cases where the HLA allele in query needs to be estimated based on other similar HLA alleles. Colors are used to denote the clusters as annotated in Fig. 3. c, Violin plot showing the distribution of mean value of variances calculated across (up to 8) algorithms across the different anchor clusters. Lower variance indicates a better prediction consistency across the different algorithms used. d, Violin plot showing the distribution of distances across the different anchor clusters. Distance of 0 indicates that the HLA allele either did not need a closest neighbor to estimate binding or that the HLA pseudo-sequences between the two alleles were identical. e, Scatter plot showing the correlation of distance to the mean variance across different HLA alleles in our dataset are connected to each other. Each center node has training data available with the size proportional to the size of the center node. Edges connect nodes representing neighboring HLA alleles (as defined by NetMHCpan4.0). Color of each node represents the anchor cluster assigned as in Fig. 3. g, Zoomed in view of the network graph in f for the top 10 largest networks. Center HLA allele is listed on the top left of each box in bold.

of nearest neighbors. Hence, we excluded the possibility of clusters arising due to bias in algorithms and their training data and believe that allele-specific differences are the true underlying reason for our observations.

2.3.3 Protein structural analysis confirms predicted anchor results

To validate our anchor predictions, we collected X-ray crystallography structures for MHC molecules with bound peptides (Appendix 2). The 166 protein structures collected corresponded to 33 HLA alleles with the majority of them containing 9-mer peptides (8-mers: 6, 9-mers: 110, 10-mers: 39, 11-mers: 11). These structures were analyzed using two methods: 1) measuring the physical distance between the peptide and the MHC binding groove and 2) calculating the solvent-accessible surface area (SASA) of the peptide residues (Figure 2.9a,b; Methods). These
Figure 2.9. Orthogonal validation using protein crystallography structures. Orthogonal validation of predicted anchor scores utilizing X-ray crystallography structures. 

a. Schematic of analysis workflow for each HLA-peptide structure collected. For the distance metric, backbone atoms were excluded with the exception of glycine. 

b. Structural example of HLA-B*08:01 bound to peptide FLRGRAYGL (PDB ID: 3X13). 

c. Example results of 47 structures collected for HLA-A*02:01 with 9-mer peptides. Top panel corresponds to distance measurements for each position while the bottom panel corresponds to SASA measurements. 

d. Distribution of Spearman correlations calculated between distance and prediction scores (top) and SASA and prediction scores (bottom). Blue line represents each respective correlation distribution while the green line shows the distribution of Spearman correlation values obtained from randomly shuffled peptide positions.
methods were selected to validate predicted anchor positions based on the assumptions that if a certain peptide position is designated as an anchor then it is 1) more likely to be closer to the HLA molecule and 2) more secluded from solvent surrounding the peptide-MHC complex compared to non-anchor positions. This is because non-anchor peptide residues should be accessible to the TCR for recognition, thus in the peptide-MHC structures collected where a TCR is not present, peptide surface area available to the surrounding solvent roughly mimics the area that would be accessible by the TCR. Thus we expect an inverse correlation between our anchor prediction scores and the distance/SASA metrics. HLA-A*02:01, shown as an example, was found to have the greatest number of qualifying structures, with 47 of them containing a 9-mer peptide (Figure 2.9c). These x-ray crystallography structures each capture a snapshot of a dynamic protein structure in constant movement. By overlaying the distance and SASA scores across all 47 complexes respectively, we observe that positions 2 and 9 are the ones most consistently close to the HLA molecule while also being secluded from the solvent. This observation corresponds well with our prediction of strong anchors at both positions 2 and 9 for HLA-A*02:01 and a 9-mer peptide. We also performed a targeted analysis on HLA-B*08:01 (2W-3W-5W-9M; blue cluster) with the limited data available and observed that positions 6 and 7 consistently bulged out whereas other positions tend to be closer to the HLA molecule while also being secluded from solvent (Figure 2.10).

To evaluate how the distance and SASA metric correlates with our prediction results across different HLA alleles, we calculated Spearman correlations between our prediction scores and distance/SASA results for each peptide position across 87 PDB structures. These structures were determined by randomly selecting at most 5 structures per HLA-length combination. The
distribution of these correlations was compared to that of a randomized dataset where positions

![Graph showing distances and SASA of peptide]

**Figure 2.10. Analysis of crystallography data for HLA-B*08:01 and 9-mer peptides.** Results of two structures produced for HLA-B*08:01 with 9-mer peptides (blue and green lines). Top panel corresponds to distance measurements for each position while the bottom panel corresponds to SASA measurements. X-axis represents positions 1 to 9 of the peptides included.

of the peptide were randomly shuffled (**Figure 2.9d; Appendix 2; Methods**). Two sample t-tests showed the distributions were significantly different from the randomized dataset with statistical values of -9.9795 (p value = 1.3757e-18) and -14.7322 (p value = 8.7472e-30) for distance and SASA metrics respectively. Results show that 91.95% of our prediction scores are inversely
correlated with the distance metric and 100% of them are inversely correlated with the SASA scores. Furthermore, we analyzed 61 protein structures that contained both the peptide-MHC complex and an additional binding TCR molecule. The distance between the TCR and the peptide showed high correlation with our prediction scores (Figure 2.11). Two-sample t-tests showed significant differences between the randomized dataset and both the HLA-peptide distance \( (p = 8.8023 \times 10^{-08}) \) and the TCR-peptide distance \( (p=2.4915 \times 10^{-13}) \). These results together strongly suggest that our anchor prediction workflow is accurately predicting allele specific anchor sites.

**Figure 2.11. Distribution of Spearman correlation values comparing anchor predictions and peptide-HLA/TCR distance measurements** This analysis is based on a collection of 61 structures obtained from PDB where a specific peptide, MHC and TCR were crystallized as a three part complex. The distribution of Spearman correlation values calculated by comparing our anchor scores to distance metrics obtained from these structures is shown. The blue line represents correlations between the predicted anchor score and the HLA-peptide distance measurements (expected to be negatively correlated because a position with a strong/high anchor score will tend to be closer to the MHC groove). The red line represents correlations between the predicted anchor score and the TCR-peptide distance measurements (expected to be
positively correlated because positions acting as an MHC anchor will tend to be further away from the TCR interface). The green line represents the correlations between the predicted anchor score and randomized distances.

2.3.4 Experimental validation shows similar anchor trends as our prediction across distinct HLA alleles

In order to further validate the predicted anchor patterns we observed, we selected a range of HLA alleles representing these patterns for experimental validation. We used two different experimental validation methods across our peptide-HLA combinations: in vitro IC50 binding assays and cell-based stabilization assays (Methods). Validations were performed on a total of 136 peptide-HLA combinations across 8 different HLA alleles selected to represent varying anchor patterns. For each HLA allele selected, we first validated peptides that were predicted to be strong binders. These peptides were originally MT peptides from either clinical datasets or TCGA samples paired with matching HLA alleles from the patient. For a validated strong binding peptide-HLA combination, we then synthesized peptides mutated in multiple ways at different positions, which each had a varying predicted score of acting as an anchor. These mutated peptides were then evaluated experimentally for MHC binding and/or stability and compared to the original strong-binding peptide (Appendix). For example, we performed an in-depth analysis for HLA-B*07:02 with a strong binding 9-mer peptide RPDVKHSKM. Overall for 9-mer peptides, HLA-B*07:02 was predicted to have a high anchor score at position 2 and a lower anchor score at position 9 (2S-9W) (Figure 2.12a). Following our computational prediction workflow, we also calculated the specific anchor trend for peptide RPDVKHSKM, which agrees with the overall 9-mer trend with positions 2, 9 and 1 as anchors in descending scores (Figure 2.12b). We mutated positions 1, 2, 5 and 9 to four different amino acids, each
Figure 2.12. Experimental Validation of anchor pattern for HLA-B*07:02. Validation results for HLA-B*07:02 with a predicted 2S-9W anchor pattern using peptide sequence RPDVKHSKM. 

a, Overall anchor prediction scores for each position of 9-mer peptides when binding to HLA-B*07:02. Higher scores indicate a higher probability of acting as an anchor. 

b, Binding affinity changes (y-axis) plotted for each position of the specific 9-mer peptide RPDVKHSKM. Each peptide position was mutated to 19 other amino acids to evaluate influence on binding affinity. 

c, MFI values measured from cell stabilization assays are plotted for both
unmutated and mutated peptides. Peptides are marked by their mutation positions (P1, P2, P5 and P9), predicted binding affinity values, amino acid changes (color coordinated with figure 5b), and mutation category (shape coordinated with figure 5d). d. Predicted binding affinity scores (log10[nM]) plotted against measured binding affinity values (log10[nM]) from IC50 binding assays. Binding categories based on measured binding affinity values are marked using horizontal lines. MFI values are overlaid using heatmap coloring for all data points. Peptide categories are marked using different shapes (coordinated with figure 5c). All non-binders were plotted with a measured binding affinity value of 7.

with varying predicted influence on the binding affinity, and measured their stability with regards to HLA-B*07:02 (Methods; Figure 2.12c). For non-anchor position P5, all mutated peptides remained strong binders. For the two weak anchor positions P1 and P9, certain amino acid changes disturbed the binding while others had little influence. For the strong anchor position P2, all amino acid changes led to a non-binder. We further performed IC50 binding assays and observed a similar trend to that of the stability experiments (Figure 2.12d). Similar mutation experiments on HLA-B*08:01, HLA-A*68:01 and HLA-A*23:01 were conducted (Figure 2.13 and 2.14). These additional experiments confirmed the varying strengths of individual positions acting as anchors for MHC alleles comprising anchor patterns 2W-3W-5W-9M, 2W-9S and 9S.
Figure 2.13. Additional experimental validation data for the predicted HLA-B*08:01 anchor pattern. Validation results for HLA-B*08:01 with a predicted 2W-3W-5W-9M anchor pattern using peptide TLFMREHNL. a, Overall anchor prediction scores for each position of 9-mer peptides when binding to HLA-B*08:01. b, Binding affinity changes (y-axis) plotted for each position of the specific 9-mer peptide TLFMREHNL. Each position of peptide was mutated to 19 other amino acids to evaluate influence on binding affinity. c, IC50 values measured from binding affinity assays are plotted for both unmutated and mutated peptides. Peptides are marked by their mutation positions (P2, P3, P5, P6 and P9), predicted binding affinity values. Lower IC50 values correspond to stronger binding between the peptide and HLA allele.

Figure 2.14. Additional experimental validation data for HLA-A*23:01 and HLA-A*68:01
a, Validation results for HLA-A*68:01 with a predicted 9S anchor pattern using peptide ELAKHACPR. MFI values measured from cell stabilization assays are plotted for both unmutated and mutated peptides at 100 nM concentration. b, Validation results for HLA-A*23:01 with a predicted 2W-9S anchor pattern using peptide QWLQPEAHF. MFI values measured from cell stabilization assays are plotted for both unmutated and mutated peptides at 20 nM concentration. Peptides are marked by their mutation positions (P1, P2, P5, P9), and predicted binding affinity values. Higher MFI values correspond to stronger binding between the peptide and HLA allele.
Overall, across 136 peptide-HLA combinations, experimental validation results were consistent with the average predicted binding affinities calculated from 8 prediction algorithms. For peptides validated using IC50 binding assays, reasonable correlations were observed between the predicted and measured IC50 values, with respect to individual HLA alleles (Figure 2.15a). When comparing the measured binding category (determined by the measured IC50) with predicted binding affinities, we saw a clear trend with high affinity binders having the lowest predicted IC50 values (Figure 2.15b). A few outliers were observed where peptides were predicted as strong binders but when validating, these peptides were categorized as non-binding. This may be explained by difficulties in manufacturing and/or solubilizing these peptides, preventing detection in our validation experiments. As expected, an inverse correlation was observed between the MFI average values and the predicted binding affinities (Figure 2.15c). Similarly, peptides with measured high affinities showed the highest MFI values. Peptides with measured binding affinities among the “medium”, “low”, “very low” and “no binder” categories were similar in their range of MFI values, indicating a difference in the sensitivity between the IC50 binding assays and cell stabilization experiments (Figure 2.15d; Methods).

We further examined how each prediction algorithm performed when compared to the measured IC50 binding values (Appendix 2; Figure 2.16). Out of the eight algorithms used, HLA-A*02:01 had the highest correlation scores for 6 out of the 8 algorithms, consistent with our observation that it has the largest training dataset (Appendix 2) and thus was expected to yield the best accuracy. Overall, NetMHCpan came out on top, showing the highest correlation values in all 4 HLA alleles where sufficient data was available (Figure 2.17). These experimental validation results combined help show the distinct anchor trends that exist among HLA alleles.
Figure 2.15. Summary of experimental validation results across all HLA allele-peptide combinations. We performed two types of experimental validations including cell-based stabilization assays (MFI values) and IC50 binding assays (log[nM]) for a total of 136 peptide-HLA combinations. The results from these assays are compared between each other and against the predicted binding affinity (averaged across 8 different algorithms). Measured binding categories are determined based on the measured IC50 values. 

a. Predicted binding affinity values plotted against the measured binding affinity (log[nM]). HLA-A*02:01, HLA-A*08:01, HLA-B*07:02, and HLA-A*02:01 were subject to linear fitting with $R^2$ values shown. HLA-A*24:02 was excluded due to the limited range of data available. Non-binding peptide-HLA combinations that have no exact measured binding affinity value available were plotted at y=7. These data points were not included when performing linear regression.

b. Measured binding categories plotted against predicted binding affinity values.

c. Predicted binding affinity values plotted against average MFI values (measured at 100 nM concentration).

d. Measured binding categories plotted against MFI average value (measured at 100 nM concentration).
Figure 2.16. Breakdown of predicted binding affinity values versus measured binding affinity by individual algorithms. Binding affinity values predicted by each individual algorithm plotted against the measured binding affinity (log[nM]). HLA-A*68:01, HLA-B*07:02, HLA-B*08:01, HLA-A*02:01 were subject to linear fitting with R2 values shown (HLA-A*24:02 was excluded due to the limited range of data available). Non-binding peptide-HLA combinations were also excluded from this plot since they have no measured binding affinity value available. Prediction scores are shown for the following 8 different algorithms: MHCflurry, MHCnuggetsI, NetMHC, NetMHCcons, NetMHCpan, Pickpocket, SMM, and SMMPMBEC.
Figure 2.17 Correlation Scores between individual MHC binding algorithm predictions and measured binding affinities across HLA alleles. Bar plot showing Pearson correlation scores calculated between predicted binding affinity values from 8 different algorithms and the measured IC50 binding affinity values for individual HLA alleles. Plots are grouped by the specific HLA allele and individual algorithms are marked with a color legend.

2.3.5 Neoantigen prioritization results are influenced by accounting for anchor locations

Current pipelines fail to take into account HLA allele-dependent effects on anchor locations and researchers lack specific tools and databases to make use of such information. While the decision of whether a neoantigen should be prioritized over others involves many aspects not discussed in detail here (including variant allele frequencies, gene expression, and manufacturability considerations to name a few), we used a straightforward approach to evaluate the effects of introducing improved anchor information on neoantigen prioritization. Interpretation of a strong binding MT peptide candidate may depend on other factors such as WT binding affinity, agretopicity, mutation position, and anchor location(s), leading to different choices when prioritizing neoantigens (Figure 2.18a). If the mutation is not at an anchor location, regardless of the WT peptide binding affinity, the MT peptide should be prioritized. In this case, the sequence
Figure 2.18. Impact of anchor position information on neoantigen prioritization decisions

a, Illustration of different scenarios that could be encountered when prioritizing neoantigens. Each circle represents a peptide residue with wild type (WT) residues indicated by orange and mutated (MT) residues by red. Anchor locations of the MHC are marked in green while TCR recognition sites are in blue. Predicted binding affinities of the MT/WT peptides are indicated using a yellow density field where higher density represents strong binding, and lower density represents weak binding. For putative strong binding MT peptides (IC50 < 500nM), the four different scenarios are depicted with the proposed varying classification (accept/reject). The classification in each scenario depends on the WT binding and location of the MT residue with respect to predicted anchor position(s). Given weak WT binding (IC50 > 500nM), regardless of the mutation’s anchor status, a strong MT binder is accepted as a neoantigen candidate (scenarios 1 & 4). However, given a strong WT binder, a peptide is accepted in the case where the MT residue is not at an anchor (scenario 2) and rejected when the MT residue is at a predicted anchor (scenario 3) since these peptides may be subject to central tolerance. b, Upset plot showing the number of intersecting peptides based on those prioritized with no anchor filter (binding affinity < 500 nM and agretopicity > 1), conventional filter (filtering based on conventional anchor assumptions) or allele-specific filter (filtering based on our computationally predicted anchor locations). Samples included for analysis were chosen such that HLA alleles were balanced appropriately. Peptides characterized differently between no anchor/conventional filter and allele-specific filter were categorized into false negatives (green circle) and false positives (red circle) with the assumption that the allele-specific filter produced more accurate results. c, Bar plot showing the percentage difference among accepted neoantigen candidates based on filters discussed in panel b. The filters in the legend represent how each starting dataset was filtered: 1) filtered by a strong binding cutoff of either 500 nM, 100 nM or 50 nM and 2) filtered based on the anchor patterns of the corresponding HLA allele. Our HLA allele filter excluded all peptides binding to HLA alleles with a canonical anchor pattern (e.g. [2,9] for 9-mer peptides). d, Examples of false positive and false negative peptides from each of the four subsets as marked in panel b. Matching HLA allele, peptide sequence, mutation position (red), median WT/MT IC50 values and fold changes are shown accordingly. Two sets of anchor locations are depicted for each scenario using semi-circles: conventional anchors are marked with light blue and allele-specific anchors are marked with dark blue. Positions where the two sets of anchors overlap are marked with split coloring of the semi-circle.

for TCR recognition contains a mutation and will not be subject to central tolerance (Figure 2.18a; scenario 1,2). Additionally, if the WT peptide is a weak binder, the MT peptide should be accepted regardless of whether the mutation is at an anchor location since both the MT and WT sequences have not previously been exposed to the immune system and therefore not subject to tolerance (Figure 2.18a; scenario 4). However, if the WT binds strongly, regardless of
agretpicity, and the mutation is at an anchor location, then this neoantigen will likely be subject to central tolerance and should be rejected from prioritization (Figure 2.18a; scenario 3). These scenarios are considered when performing the following anchor position impact analysis.

Our cohort impact analysis involved an additional set of TCGA patient samples where neoantigens were predicted for 923 selected patient-HLA allele pairings. When selecting patient-HLA paired samples we chose from a balanced HLA allele distribution (Methods). Our intent was to give a more balanced view of the impact across all HLA alleles without overt bias for the most common alleles. We first examined the proportion of SNV-induced neoantigens that went into each of the 4 scenarios described in Figure 2.18a and found that 17%, 57%, 8% and 18% of all peptide candidates fell into scenarios 1, 2, 3 and 4 respectively (Figure 2.19). All potential neoantigens were filtered according to three different criteria: A) mutant IC50 < 500nM and agretpicity > 1 (‘no anchor filter’), B) supplementing this with a conventional anchor.
assumption (‘conventional filter’), or C) using our computationally predicted anchor locations (‘allele-specific filter’) (Methods). Peptide results showed that under the no anchor filter 57.9% of neoantigens are accepted compared to 93.2% under the conventional filter and 94.8% under the allele-specific filter, showing an overall net gain in the number of peptides when taking anchor considerations into account. When comparing filtered data sets under different criteria, 38.3% of neoantigens are potentially misclassified using the no anchor filter, and approximately 5.7% of candidates are potentially misclassified between the conventional filter and the allele-specific filter (Figure 2.18b). These misclassifications involve the inclusion of peptides that are likely to be subject to tolerance (and could lead to false positives) and exclusion of peptides that could be strong candidates (false negatives). We repeated this analysis to see if our observations are consistent when 1) using different binding affinity cutoffs for strong-binding peptides (< 100 nM and < 50 nM) and 2) looking exclusively at peptide-MHC pairings that have a non-conventional anchor pattern (Figure 2.18c; Methods). Our results showed that when comparing results between the no anchor (A) and any anchor considerations (B or C), differences of accepted candidates are similar across the four criteria (< 500 nM, < 100 nM, < 50 nM, and HLA filter). When comparing results between the allele-specific considerations (C) and the more naive approaches (A or B), we see an increase in percentage of difference with our HLA allele filtered dataset indicating a greater impact on neoantigen prioritization when using our allele-specific anchor method. This was expected since peptides analyzed in this dataset corresponded to HLA alleles with non-conventional anchor patterns. We also see a decrease in percentage when applying a binding affinity cutoff of 50 nM which we believe is in part due to less HLA alleles passing the stricter binding cutoffs (81 versus 98 unique HLA alleles found in datasets with cutoffs 50 nM and 500 nM respectively). By comparing peptides prioritized using
Figure 2.20. Patient-level analysis for impact of anchor considerations on neoantigen prioritization. Patient-level impact of anchor position information on neoantigen prioritization decisions using 100 randomly selected samples from a pool of 1,356 TCGA samples. 

a, The distribution of tumor sample types for a TCGA sample pool (n = 1,356) is shown as a bar plot. The x-axis represents the different cancer types included and the y-axis shows the number of patient samples. 

b, A scatterplot shows the number of unique variants plotted against the number of predicted strong binding neoantigen candidates. Neoantigen candidates were compiled such that each variant had its top neoantigen candidate selected. The candidates were subsequently filtered based on a 500 nM binding affinity cutoff for each patient. The inner bar plot shows the distribution of tumor types for the randomly selected 100 patients. The legend shows the color labeling for each tumor type and is consistent between outer scatterplot and inner bar plot. 

c, d, e: Histogram plots showing distribution of neoantigen prioritization decision differences between filters A, B and C (Methods). Differences were normalized with respect to each individual patient’s neoantigen counts and presented as the percentage of peptides that would be classified differently under each anchor interpretation scheme in terms of its neoantigen candidacy. The x-axis shows the range of percentage differences between filters while the y-axis shows patient sample counts for respective bins.
the allele-specific anchor filter and those from the no anchor/conventional anchor filters, we
highlighted the potential sources for false positives and false negatives (Figure 2.18b). Examples
of each scenario were pulled from our dataset to show how peptides passed or failed individual
filters (Figure 2.18d).

We additionally performed a patient-level impact analysis using 100 randomly selected
TCGA samples, and predicted neoantigens each with their full set of class I HLA alleles (up to 6)
(Methods; Appendix; Figure 2.20a,b). The neoantigen candidates were prioritized using the
same set of criteria applied in the previous cohort analysis. We observed a significant impact on
neoantigen prioritization results depending on the chosen filtering criteria. Specifically between
the no anchor filter and the allele-specific filter, 99% of the patients analyzed had at least one
neoantigen decision changed, with a median of 11 peptides with altered decisions, per patient.
Similarly, between the no anchor filter and the conventional filter, 98% of the patients analyzed
had at least one decision changed (median: 11 peptides) and between the conventional filter and
the allele-specific filter, 65% of the patients had at least one decision changed (median: 1
peptide) (Figure 2.20c,d,e). These results show the potential widespread effect of anchor
considerations on patient-level prioritization results.

2.4 Methods

2.4.1 Input data for identifying strong binding seed peptides for anchor site prediction

We assembled peptide data from various sources where binding prediction data were
available through clinical collaborations and supplemented these with TCGA datasets where
necessary to achieve better representation of less common HLA alleles. Datasets from clinical
collaborations that were incorporated include 7 triple-negative breast cancer samples, 54
lymphoma samples, 20 brain tumor samples, and 6 melanoma samples. Additionally, we mined data from 9,216 TCGA samples to optimize the number of strong binding peptides matched to each HLA allele by adding 10 samples for insufficient (<10 strong binding peptides) and 15 samples for previously unseen HLA alleles. Of these, 1,356 TCGA samples were used to generate seed HLA-peptide combinations to be used for downstream simulations. High quality variants included from TCGA samples were obtained from the Genomic Data Commons and selected according to their filter status (pass only) and required to be called by at least 2 out of 4 variant callers as previously described. Peptide lengths considered ranged from 8- to 11-mers. All selected data samples were run through pVACseq with the following options with all available class I prediction algorithms: --e 8,9,10,11, --iedb-retries 50, --downstream-sequence-length 500, --minimum-fold-change 0, --trna-cov 0, --tdna-vaf 0, --trna-vaf 0, --pass-only. In total, these datasets corresponded to 1,443 tumor samples, representing 328 matching HLA alleles, with 737 of these having more than 10 strong binding peptides, and a grand total of 609,807 strong binding peptides for use in the following analyses (Figure 2.2; Appendix).

2.4.2 Computational prediction of anchor site positions for 328 HLA alleles

Peptides collected from input datasets were first filtered for strong binders using a binding affinity cutoff of 500 nM. These were used to build a seed dataset consisting of peptides predicted to be strong binders to individual HLA alleles. We first performed a saturation analysis to determine the appropriate number of random peptides needed to obtain a robust estimate of the likely anchor site locations of each HLA allele. This was done using peptides collected for HLA-A*02:01, where over 1,500 peptides were obtained for each peptide length. Random sampling with a subset size of 10 peptides showed consistently high correlation (> 0.95) with the largest subset size used where all 1000 peptides were incorporated (Figure 2.7).
downstream analysis, for each unique HLA and peptide length combination, 10 peptides were randomly selected from the database. For each of the 10 starting peptides at each position n, we obtained a score that reflects how much a mutation at this position (by mutation to all 19 other amino acid identities) will affect the overall binding affinity:

\[
(\text{abs}(Y_{(n,1)}-X)+...+\text{abs}(Y_{(n,20)}-X))
\]

where X is the binding affinity of the unmutated peptide, and \(Y_{(i,j)}\) is the binding affinity of the peptide mutated at position i to amino acid number j (total of 20 possible amino acids to mutate to). All binding affinities were calculated using pVACbind from pVACtools (version 1.5.0)\textsuperscript{164} in which the following algorithms were selected: NetMHC\textsuperscript{114}, NetMHCpan\textsuperscript{182}, MHCnuggets\textsuperscript{183}, MHCflurry\textsuperscript{102}, SMM\textsuperscript{111}, Pickpocket\textsuperscript{113}, SMMPMBEC\textsuperscript{112}, NetMHCons\textsuperscript{184}. The median binding affinity across all 8 of these algorithms was used both to nominate strong binder peptides for the seed dataset, and to assess the impact of in silico mutation at each position of these peptides. For each strong binding peptide in the seed dataset, this systematic in silico mutation led to [length x amino acid identity x algorithm] binding predictions (e.g. 1,368 binding predictions for a single 9-mer peptide).

Each position was assigned a score based on how much binding affinity values were influenced by mutations at that position. These scores were then used to calculate the relative contribution of each position to the overall binding affinity of the peptide. Positions that together account for 80% of the relative overall binding affinity change were assigned as anchor locations for further impact analysis.
2.4.3 Evaluating different seed peptide sources and their impact on the anchor patterns predicted

Three additional sources were explored and compared to peptides from our seed dataset to investigate whether the source of these strong binding peptides would influence our results. For comparison, we selected 1,000 strong binding peptides (median predicted binding affinity < 500nM) from the following four sources: randomly generated peptides, peptides randomly selected from the WT proteome, peptides randomly selected from a viral genome, and peptides from our seed dataset. To obtain randomly generated peptides, we first generated a random list of amino acids (100,000,000 aa in length) using a random choice function and tested 50,000 9-mer peptides to reach 1000 predicted strong binders. For peptides generated from the WT human proteome (ensembl) and the viral proteome (variola virus), we randomly selected 30,000 9-mer peptides from proteins with valid sequences longer than 9 amino acids.

For each strong binding peptide, we followed our computational workflow for anchor predictions and mutated each position of the peptide to all possible amino acids and accessed the change to binding respectively. Normalized scores across all peptide positions for 1,000 peptides were plotted in Figure 2.7.

2.3.4 Bias analysis for HLA allele-specific anchor patterns

To evaluate whether our anchor patterns are influenced by the training data available for each HLA allele, we downloaded the training dataset of NetMHCpan4.0. Additionally, for HLA alleles with no training data available, we determined which HLA allele was its closest neighbor defined by NetMHCpan4.0 and what the distance measurements were between the neighboring alleles. With these data obtained, we used CytoScape to visualize the relationships between all alleles using network graphs. Each center node represents an HLA allele with training data (size
of dataset correlates with the size of each node) and each connected node represents a
neighboring HLA allele that uses the center node allele for its binding estimations. Distances are
marked on the edges of each graph. Colors of each node reflect the anchor cluster assigned as
shown in Figure 2.5.

2.4.5 Input data for orthogonal evaluation of predicted anchor sites

To evaluate our anchor predictions, we collected 166 protein structures (pdb format) of
peptide-MHC complexes and 61 peptide-MHC-TCR complexes from the Protein Data Bank\textsuperscript{188} by querying for structures containing macromolecules matching class I HLAs. Structures were
additionally reviewed to ensure valid peptide length (8-11) and those with TCRs attached were
separated into a different list for downstream analysis to allow accurate solvent-accessible
surface area (SASA) calculations. The HLA-peptide structures corresponded to 33 HLA alleles
with peptides of varying lengths (8 to 11mer), while the HLA-peptide-TCR structures
corresponded to 12 HLA alleles. A complete list of PDB ids selected for this analysis can be
found in Appendix

2.4.6 Orthogonal validation of predicted anchor sites by analysis of pMHC structures

The structures of peptide-MHC molecules were analyzed to infer potential anchor
locations/residues. All PDB structures were analyzed in python using the MDTraj package\textsuperscript{189}. For each position of a peptide bound to an HLA, we utilized two different metrics: 1) minimum
distance of non-backbone atoms to all HLA associated atoms and 2) estimated solvent-accessible
surface area (SASA) of the residue. In method 1, we calculated the distances between each atom
of each residue and all HLA associated atoms. Non-backbone atoms were ordered by their
distance to the closest HLA-associated atom and the top 50% were used to calculate an average
distance representing an entire residue (with the exception of glycine where all atoms were considered). In method 2, we directly calculated the SASA of each residue (shrake_rupley function in MDTraj), which was used to infer the likelihood of being able to be recognized by the T-cell receptor. After calculating these values for each position of the peptide, they were directly compared to the anchor prediction scores by calculating a Spearman correlation. In the case of the distance metric, we expect positions of the peptide closer to the MHC to be more likely an anchor and those further (“bulging out”) to more likely interact with the TCR. Similarly, for the SASA metric, if a peptide position is more solvently accessible (higher SASA value) we expect it to be more accessible to the TCR as well and those that are less accessible would be more likely interacting with the MHC as an anchor.

For an overall evaluation of how well our anchor predictions correlated with these metrics (distance and SASA), Spearman correlations were determined across all structural data collected. For example, for a 9-mer peptide, a Spearman correlation was calculated for the 9 anchor prediction scores from the in silico mutation exercise compared to the 9 distance or SASA estimates obtained from the structure analysis. Out of 166 peptide-MHC structures collected, correlation values for 87 were plotted by randomly selecting at most 5 structures per HLA-length combination (Appendix). For comparison, we also randomly shuffled distance and SASA scores across all positions of individual peptides and calculated correlation scores against this randomized dataset. The different sets of correlation values were then fit to Gaussian distributions (Figure 2.9). Non-paired two sample t-tests assuming unequal variance were performed to evaluate the differences among distributions (using python SciPy scipy.stats.ttest_ind).
Additional analysis was performed on the 61 peptide-MHC-TCR structures collected. After randomly selecting at most 5 structures per HLA-length combination, Spearman correlations derived from 31 structures were plotted. Correlations were calculated for 1) distance from peptide to HLA versus anchor prediction scores and 2) distance from peptide to TCR versus anchor prediction scores. Once again, the HLA-peptide distances were randomly shuffled and used as comparison and two sample t-tests were performed to evaluate the differences among distributions.

2.4.7 Input data for evaluating the impact of anchor site considerations

To evaluate how anchor site considerations might influence neoantigen prioritization decisions, we considered a balanced HLA allele distribution when selecting input data. We randomly sampled up to 10 corresponding TCGA samples for each HLA allele with sufficient data (at least 3 out of 4 lengths have 10 or more matching peptides). 923 TCGA-HLA combinations were chosen from a total of 9,216 TCGA samples excluding the 1,356 used for the seed anchor site prediction data set described above. The 923 TCGA-HLA combinations corresponded to TCGA patients (Appendix). To further evaluate impact of anchor considerations on a patient-specific level, an additional 100 TCGA patients were selected from the original 1,356 TCGA patient samples where we had neoantigen predictions for the patient’s full set of HLA alleles (Appendix).

2.4.8 Evaluating the impact of anchor site consideration on neoantigen prioritization

To analyze the importance of positional information on prioritization of neoantigens, TCGA patient samples were used as input and run through pVACtools (version 1.5.2) using the following options: -e 8,9,10,11, --iedb-retries 50, --downstream-sequence-length 500, --
minimum-fold-change 0, --trna-cov 0, --tdna-vaf 0, --trna-vaf 0, --pass-only. The neoantigen candidates were then filtered and prioritized according to different criteria: A) Basic Filter: mutant peptide IC50 < 500 nM and agretopicity > 1, B) Decision based on a conventional anchor assumption that anchors are located at position 2 and the C-terminal position, C) Decision based on computationally predicted anchor locations. Specifically, under filter A) the position of a mutation with respect to MHC anchor positions is not considered. An accepted peptide means: MT peptide IC50 < 500 nM and WT IC50 / MT IC50 > 1, otherwise the peptide is rejected. Under filter B), anchor positions are defined to be 2 and n for all n-mer peptides. Under filter C), anchor positions are defined by our computational predictions, which are allele-specific. For both filters B and C, peptides are accepted if 1) the MT IC50 < 500 and WT IC50 > 500, or 2) the MT IC50 < 500, WT IC50 < 500 and the mutation is at a non-anchor location, as defined by the anchor definition of filters B and C respectively.

For filter C, anchor positions were defined individually for each peptide using the following strategy: Per anchor calculation results from our computational workflow, each position of the n-mer peptide was assigned a score based on how binding to a certain HLA allele was influenced by mutations. These scores were then used to calculate the relative contribution of each position to the overall binding affinity of the peptide. We ranked the normalized score across the peptide in descending order (e.g. [2,9,1,3,2,8,7,6,5] for a 9-mer peptide) and started summing the scores from top to bottom. Positions that together account for 80% of the overall binding affinity change (e.g. 2,9,1) were assigned as anchor locations for impact analysis. Assuming \( P_n \) represents the normalized score of position \( n \) within the peptide, for each HLA allele, the anchor(s) is determined as following:

\[
\sum_{a}^{n} P_a \geq 0.8 \text{ where } P_a > P_b \ldots > P_n
\]
Filtered lists were then compared for overlap and differences. We also followed the same evaluation process for different starting candidate lists of varying stringency, including: candidates filtered with a strong-binding cutoff of 100 nM, candidates filtered with a strong-binding cutoff of 50 nM and candidates filtered by their binding HLA allele’s anchor patterns. For our HLA allele filtered dataset, all HLA alleles with an exact [2, n] anchor pattern (for n-mer peptides) were considered canonical and excluded from further evaluation.

For our cohort analysis, all neoantigen candidates were considered with no additional filtering. For our patient-level analysis, neoantigen candidates were processed additionally using the top_score_filter (“pVACseq top_score_filter” command of pVACtools) to generate top neoantigen candidates for individual variants. These top candidates were compiled and the same filters A, B, and C were used to determine prioritization decisions. The percentage differences between filters were calculated based on decisions for all top candidates for each individual patient.

2.4.9 HLA coverage and population frequency

Global HLA allele frequencies were generated using data from the Allele Frequency Net Database. The database contains HLA genotype data for Class I alleles across 197 distinct populations. Two populations in the database (“Chile Santiago” and “Russia Karelia”) did not have ambiguity-resolved HLA genotype data and were excluded from this analysis. Global HLA allele frequencies were calculated by (1) aggregating all 195 sample populations, (2) summing HLA allele counts over all sample populations, and (3) dividing HLA allele counts over total counts of the HLA gene across all populations. It should be noted that the HLA frequencies calculated do not reflect true global HLA frequencies since true population/region sizes were not considered. To calculate the percentage of population that our 328 HLA alleles affect, Class I
alleles were split into respective subclasses of HLA-A, HLA-B and HLA-C. Global frequencies were summed in each subclass to obtain the percentage of population potentially affected by our HLA allele anchor results.

2.4.10 Selection of validation peptides

In order to select a group of peptide-HLA combinations for validation experiments, we first examined our seed database of predicted strong binding combinations of peptides and HLA alleles. The database was then filtered for preselected HLA alleles with varying anchor patterns. For each HLA allele within an anchor subgroup, we then selected 3-5 peptides and performed validation experiments to measure their binding affinity. Validated peptides were then mutated at various positions (including those predicted to be anchors and non-anchors) to evaluate amino acid changes and their effect on binding affinity. Due to the limited resources and high cost of synthesizing peptides and performing validation experiments, we strategically chose 4 different amino acid mutations for each position evaluated. Both positions and amino acid mutations were chosen to optimize informativeness. The amino acid mutations were selected based on their predicted influence on binding affinities by two methods. For our first batch of peptides and their in-depth analysis, we selected two of the most/least disturbing mutations each according to our predictions. For our second batch of peptides, this strategy was further optimized to select 1 mutation from each quartile based on predictions of how much they disturb the binding interaction. These mutated peptides were then synthesized by GenScript (Piscataway, NJ, USA) for further validation. The peptides were ordered with the following specifications: quantity: 4 mg, weight: gross, purity: ≥75%, delivery format: lyophilized, aliquoting to vials: 4.
2.4.11 Peptide:MHC IC50 binding assays

As previously described\(^\text{190}\), peptide binding validation assays to determine IC\(_{50}\) values were carried out by Pure Protein LLC, utilizing the method of fluorescence polarization (FP). The technique is unique among methods used to analyze molecular binding events because it allows the instantaneous measurement of the ratio between free and bound labeled ligands in solution without any separation steps. FP is based on the principle that if a fluorescent-labeled peptide binds to the soluble HLA molecule of higher molecular weight, polarization values will increase due to the slower molecular rotation of the bound probe. In this competition assay, a reference fluorescent-labeled peptide is incubated with activated sHLA in the presence of the neoantigen-derived peptide competitor and peptide/HLA interaction is monitored over time. A positive response occurs when the peptide of interest outcompetes the labeled peptide tracer. A negative response will take place when the peptide of interest has no binding characteristics and only the tracer is assembling with the sHLA. Competition experiments were analyzed by plotting FP values as a function of the logarithms of competitor concentrations. The binding affinity of each competitor peptide was expressed as the concentration that inhibits 50% binding of the FITC-labeled reference peptide. Observed half-maximal inhibitory concentrations (IC\(_{50}\)) were determined by nonlinear curve fitting to a dose-response model with a variable slope using the specific software Prism (Graph Pad Software, Inc., San Diego, CA). In order to prioritize epitopes with greatest potential, ranked peptide IC\(_{50}\) values were classified as log-dependent values (x) based on preset affinity categories as described in Buchli et al. \(^\text{2005}\), into high (x<3.7), medium (3.7 < x < 4.7), low (4.7 < x < 5.5), very low (5.5 < x < 6.0) and no binder (x > 6.0). Binding assays were performed for all (or part of) the peptides with the following matching HLA alleles: HLA-B*07:02, HLA-B*08:01, HLA-A*02:01, HLA-A*24:02, and HLA-A*68:01.
2.4.12 Cell-based peptide:MHC stabilization assays

The recipient of all HLAs of interest in this study was a TAP deficient, class I negative cell line, that was created by introduction of HSV-ICP-47 (gift from Ted Hansen) using the pMIP (puro<sup>r</sup>) retrovirus and selection of puromycin resistant cells into the Class I negative, lymphoblastoid cell line, 721.221 (gift from M. Colonna, Washington University Saint Louis; also available from Millipore Sigma, catalog number SCC275) cultured in Iscoves MEM, 10% (v/v) FBS with 0.6 µg/mL puromycin. HLA cDNA (IDP-IMGT/HLA) was prepared synthetically (Blue Heron, Bothell, WA) and shuttled into pMIG (GFP) retroviral vector. All HLA expressing cells were enriched by sorting GFP positive cells to greater than 95% using a Sony Synergy Flow Cell Sorter. For peptide stabilization assays, the cell line expressing the HLA, was washed twice in PBS and then serum starved for 1 hour in RPMI1640, 10 mM HEPES, 1x non-essential amino acids, 1 mM sodium pyruvate, 2 mM glutamine, prior to plating, washed twice and incubated with 100 µg/mL peptide of interest previously solubilized in 10% DMSO and sterile filtered through a 0.2 µm centrex filter. The cell suspensions were maintained at ambient temperature for 1.5-3 hours and then shifted to 37°C overnight at 5% CO<sub>2</sub>. To quantitate peptide stabilized complexes, the cell suspensions were then washed twice to remove unbound peptide and then incubated with APC conjugated W6/32 Mab (Biolegend, 311410) for 30 minutes at ambient temperature. Cells were washed twice and complexes were detected in the presence of 7-AAD to remove dead cells from the analysis. The median fluorescence intensity of 7-AAD negative, GFP/APC positive cells were quantitated by FlowJo v.10.6.1 after collection on a Beckman Coulter Navios three laser flow cytometer. Each HLA tested utilized a Flu viral positive control peptide known to stabilize the HLA of interest (Appendix) and a negative control peptide that stabilized a different class I molecule. Each assay was completed 2-3 times.
in duplicate (N=4-6) and graphed in GraphPad Prism v.9.2.0 after subtraction of the MedFI of the no peptide control. The corrected mean fluorescence intensity (MFI) values are included in Appendix. Stabilization assays were performed for all of the peptides with the following matching HLA alleles: HLA-A*02:01, HLA-A*23:01, HLA-A*24:02, HLA-A*31:01, HLA-A*68:01, HLA-B*07:02, and HLA-B*18:01.

2.4.13 Statistical Analysis & Code/Data Availability

Statistical analysis was performed using the python SciPy package. Non-paired two-sample t-tests assuming unequal variance were performed to evaluate the differences among distributions for Figure 2.9 T-tests were performed using the scipy.stats.ttest_ind function. The normal distribution of variables was calculated using the Shapiro-Wilk test.

Sequence data for each cohort analyzed in this study are available through the Database of Genotypes and Phenotypes (dbGaP) at the following accession IDs: phs000178 for TCGA cohorts, phs002612 for GBM, phs001229 for lymphoma and phs003153.v1.p1 for melanoma.

Code used in our prediction pipeline and analyses are organized and available at https://github.com/griffithlab/anchor_huming_et al_2023 (10.5281/zenodo.7672092).

2.5 Discussion & Conclusions

We developed a computational workflow for predicting anchor position scores for a wide range of the most common HLA alleles. Our results show that anchor positions vary substantially between different HLAs. A subset of our prediction results were confirmed by analyzing available crystallography structures of peptide-MHC complexes. We further experimentally validated HLA allele anchor patterns using binding assays and cell-based stabilization assays. The underlying quantitative scores from our anchor prediction workflow are available for
incorporation into neoantigen prediction workflows and we believe this will improve their performance in predicting immunogenic tumor specific peptides. For simplicity, our illustrations have depicted peptide residues as either anchoring or potentially participating in TCR recognition, although previous research has shown that heteroclitic peptides can alter both simultaneously\textsuperscript{173}. Hence, anchor residues and TCR recognition sites should not be considered mutually exclusive and should ideally be interpreted quantitatively where the anchor scores (provided in Appendix) reflect the probability of a peptide position participating in binding.

Using an independent pool of TCGA samples, previously excluded from the computational prediction process, we show that consideration of anchor prediction results can have a significant impact on neoantigen prioritization. In this study, the choices of whether to accept or reject a prioritization decision were based on hard cutoffs determined using an objective strategy across all HLA alleles. However, when making clinical decisions, we recommend using the actual anchor scores as guidance when prioritizing candidates. In most neoantigen characterization workflows, numerous other factors are taken into account to arrive at an overall prioritization decision, which may further increase differences between filtering strategies. Additionally, while our anchor results represent overall averaged scores across multiple neoantigens, slight variations of patterns do exist among different peptides for the same HLA allele. It would generally be computationally prohibitive to perform our workflow on a large scale for all possible neoantigen candidates. However, as a compromise, one could repeat our detailed process to generate peptide specific anchor predictions after performing other filtering strategies (e.g. genomic information including VAF and expression) and arriving at a shortlist of candidates.
Anchor results not only impact the selection process of neoantigens for personalized cancer vaccines, but also change the way neoantigen load estimation is currently defined. Neoantigen load estimation is commonly defined as the number of peptides whose binding affinity passes a certain threshold. However, this threshold, meant to limit to the approximate number of strong binding neoantigens, should also take into account the mutation position, HLA specific anchor locations, and agretopy for more precise estimation. Our anchor impact analysis demonstrates the effect of this alteration on estimation results. Moreover, our analysis results show that there is a net gain of neoantigen candidates when taking anchor considerations into account compared to the commonly used agretopy filters. This becomes important in the context of neoantigen prioritization, particularly when the minimum number of peptide vaccine candidates cannot be met for patients due to low tumor mutational burden. However, we also acknowledge that while our approach improves sensitivity it could come at the cost of slightly decreased specificity for patients with high neoantigen burden.

The neoantigen selection process requires careful consideration of numerous aspects, which have been discussed extensively\textsuperscript{163}. In general, neoantigen-based vaccines act by stimulating the patient’s immune system for the production of activated cytotoxic T cells. However, compared to viral antigens where the protein sequence is entirely foreign, neoantigens, particularly those developed from SNVs, have merely subtle differences from the individual’s wildtype proteome. Thus, the need for a WT versus MT peptide comparison, while considering anchor locations, is an aspect specific to tumor neoantigens that other vaccine development pipelines could generally ignore. Though neoantigens derived from in-frame or frameshift indels diverge more from the WT sequence and are generally less influenced by our findings, cases where such mutations are located towards the beginning or end of a neoantigen may still cause anchor disruption in an
allele-specific manner. Additionally, more work should be done to characterize the similarity of neoantigen candidates and the patient’s wildtype proteome for an overall accurate prioritization process.

Recent work, such as that from the Tumor Neoantigen Selection Alliance (TESLA)\textsuperscript{175}, have hinted at the potential importance of anchor locations. In that study, researchers made the unexpected observation that among the 37 positively validated neoantigen candidates, none of the peptides had a mutation at position 2, a common anchor position for a range of HLA alleles, despite a high number of prioritized neoantigens with a position 2 mutation. Possible explanations could include the fact that neoantigens with the mutant residue at a strong anchor position have a disadvantage over those present at TCR sites as they require their WT counterpart to be a poor binder and the threshold for determining sufficiently weak binding of a peptide is unclear. While a 500nM cutoff is widely used, it is not universally accepted and allele-specific binding thresholds continue to expand for a larger group of HLA alleles. However further investigation is required to address questions raised by such observations.

In addition to the limitations of being applicable to a subset of neoantigens derived from SNVs and certain indels, our work could be expanded to a wider range of HLA alleles. A larger HLA-peptide seed dataset could be achieved through a wide-scale prediction of strong binders for rare HLA alleles by mutating the wildtype proteome. There have also been publications noting the sequence motifs in HLA Class II binding peptides \textsuperscript{191}, as well as others using molecular docking approaches \textsuperscript{192} in an attempt to estimate anchor positions for HLA Class II alleles. However, class II alleles differ substantially from class I due to the difference in length of binding peptides and the binding pocket being more open and composed of a protein dimer. Additional data generation and further research will be required for the expansion of our workflow to address
class II, though we hypothesize that the same principles should be applicable. Furthermore, while x-ray crystallography structures show support for our anchor location predictions, experimental validation with neoantigens designed to induce T-cell activation is needed to explicitly showcase the importance of our results in clinical settings. Although numerous clinical trials using neoantigen-based vaccines are underway, results published show a low accuracy for current neoantigen prediction pipelines\textsuperscript{193}. By accounting for additional positional information, we hope to significantly reduce the number of false positive candidates and rescue false negative neoantigens to increase prediction accuracy. A prioritization strategy utilizing anchor results has been incorporated into the visual reporting of our neoantigen identification pipeline pVACseq\textsuperscript{164}.

Machine learning algorithms have been widely applied in the context of neoantigen binding predictions. However, machine learning models trained on experimentally validated data with T-cell activation results are lacking and identifying features for these models is an active area of research. Anchor location scores may serve as an additional feature in machine learning model training on clinical data. This will allow for a more nuanced approach where anchor scores may be weighted for each allele-peptide pairing accordingly. These results and tools will help streamline the prioritization of candidates for neoantigen vaccines and may aid in the design of more effective cancer vaccines.

2.6 Data & Code Availability

Sequence data for each cohort analyzed in this study are available through the Database of Genotypes and Phenotypes (dbGaP) at the following accession IDs: phs000178 for TCGA cohorts, phs002612 for GBM, phs001229 for lymphoma and phs003153.v1.p1 for melanoma. Code used in our prediction pipeline and analyses are organized and available at

2.7 Acknowledgements

We thank the patients and their families for the donation of their samples and participation in clinical trials. The results shown here are in part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

MG was supported by the National Human Genome Research Institute (NHGRI) of the National Institutes of Health (NIH) under Award Number R00HG007940. MG and OLG were supported by the NIH National Cancer Institute (NCI) under Award Numbers U01CA209936, U01CA231844 and U24CA237719. MG was supported by the V Foundation for Cancer Research under Award Number V2018-007. MG, OLG, HX, WEG and TAF were supported by the NCI under Award Number U01CA248235. TAF was also supported by the Jamie Erin Follicular Lymphoma Consortium and the Siteman Cancer Center (P30CA91842). This work was also supported in part by the Washington University Institute of Clinical and Translational Sciences from the National Center for Advancing Translational Sciences (NCATS) of NIH under Award Number UL1TR002345. MMR was supported by Washington University's Genome Analysis Training Program (GATP) from the National Human Genome Research Institute (NHGRI) of the NIH under Award Number T32HG000045. SS was supported by the Prince Mahidol Award Youth Program of the Prince Mahidol Award Foundation under the Royal Patronage of HM the King of Thailand.

2.8 Author Contributions

T.A.F., A.R., and W.E.G. contributed to acquisition and generation of patient sample data. H.X., J.H., S.K., S.P.G., T.M.J., C.A.M., W.E.G., O.L.G., M.G. contributed to study design. M.G. and O.L.G. contributed to the overall supervision of the project. H.X. and J.M. wrote the manuscript and created all figures in consultation with M.G. and O.L.G. All authors provided critical feedback and helped shape the research, analysis and manuscript.
Chapter 3: pVACview: an interactive visualization tool for efficient neoantigen prioritization and selection

3.1 Preamble
This work has been submitted to Genome Medicine and is currently undergoing revision in response to peer reviews.


3.2 Background
Neoantigens are unique peptide sequences generated from mutations in tumors. These antigens provide an avenue for tumor-specific immune cell recognition and have been found to be important targets for cancer immunotherapies. Effective neoantigens, presented by the major histocompatibility complex (MHC) and thus introduced to the patient’s immune system, can prime and activate CD8+ and CD4+ T cells for downstream signaling of cell-death. Previous studies have shown that patients with relatively high tumor mutation burden tend to have stronger responses to neoantigen based immunotherapy treatments. With the advent of whole exome and RNA sequencing technologies, researchers and clinicians are now able to computationally predict neoantigens for experimental studies of T cell biology in cancer or for
**Figure 3.1. Overall process of neoantigen generation and presentation from tumor specific mutations.** Illustration demonstrating the key steps throughout the process of neoantigen generation, processing, binding, transportation and presentation. This overview highlights critical aspects including: 1) founding clone versus subclonal tumor mutations, 2) different transcripts with varying expression, 3) proteasomal processing generating peptides of varying length from different registers, 4) potential of binding to individual patient HLA alleles, 5) mutation and anchor positions of the neoantigen when presented to T-cells.

design of personalized neoantigen therapies based on patient-specific mutations. Examples of such therapies include personalized neoantigen vaccines, TCR mimic antibodies, personalized adoptive T cell therapies and engineered T cell therapies. However, numerous aspects of the process of neoantigen generation and presentation must be considered for effective target selection (Figure 3.1). These aspects include but are not limited to: 1) neoantigen mutation identification and expression, 2) peptide processing and transportation, 3) peptide-MHC binding, 4) peptide-MHC stability, and 5) recognition by cytotoxic T cells.

There has been a rapid development of computational tools in an attempt to account for these complexities. Pipelines have been developed to allow researchers to run an ensemble of many tools for individual patients, generating more than 118 features, which include metrics such as algorithmic predictions of binding, allele frequency and expression, similarity to reference genome, and others. However, the results from these complex pipelines are often overwhelming in number, difficult to navigate, and require extensive knowledge of the underlying tools for accurate interpretation.

Though gene expression and predicted peptide binding affinities are common features of most approaches, recent studies have also shown the importance of mutation location, allele-specific anchor locations, and the variation of T-cell response to long versus short peptides. These additional complexities can be difficult to interrogate directly from computational pipeline outputs, if they are available at all. With the high cost of personalized
neoantigen therapies and difficulties in accurate neoantigen prioritization, it is critical to provide clinicians with multiple levels of information to support the most effective and efficient neoantigen target selection. As with most personalized treatments, choosing the right approach depends on marrying clinical information and observations with genomic data and algorithmic outputs. Supporting this effectively means moving beyond static spreadsheets and building dynamic interfaces that provide layered information in an intuitive manner. To address these concerns, we built a comprehensive visualization tool, pVACview, that takes complex neoantigen candidate information as input, visualizes the output with multiple levels of detail, and exports results and annotations for further review and submission. Previously, our cancer immunotherapy pipeline, pVACtools \textsuperscript{164}, was able to generate numerous features for neoantigen characterization. However, these outputs also required extensive additional analysis due to the intricate nature of neoantigen features. With pVACview, we now provide clinicians and researchers with a complete neoantigen detection and design pipeline. The application is compatible with human, mouse and canine data and has been used in the setting of several cancer immunotherapy clinical trials (e.g. NCT04397003, NCT03422094, NCT04015700 etc.).

3.3 Implementation

pVACview is written in R and is implemented as part of pVACtools, which is a computational toolkit that helps identify and visualize neoantigen candidates\textsuperscript{164,165}. While pVACview can be used as a stand-alone tool, we highly recommend using pVACtools to generate the required inputs to ensure compatibility. Code changes are integrated using GitHub pull requests (https://github.com/griffithlab/pVACtools/pulls). Documentation is hosted on Read the Docs (readthedocs.org) and can be viewed at

A demonstration data set is provided and consists of Class I and Class II neoantigen candidate files generated from the HCC1395 breast cancer cell line and its matched lymphoblastoid cell line HCC1395BL (please refer to data availability section). The tumor and normal datasets were processed using an immunogenomics pipeline written in CWL https://github.com/genome/analysis-workflows/blob/master/definitions/pipelines/immuno.cwl. This pipeline starts with raw tumor/normal exome and tumor RNA-seq data and performs alignment, HLA typing, germline variant calling, somatic variant calling, variant phasing, variant annotation, expression analysis, RNA fusion detection and neoantigen identification. Aggregated neoantigen reports and metrics files used as inputs to pVACview were generated using the `pvacseq generate_aggregate_report` command. These datasets are available at https://github.com/griffithlab/pVACtools/tree/latestpvactools/tools/pvacview/data.

3.3.1 Overall architecture of the software implementation

pVACview can overall be split into three main interfaces: user data upload, neoantigen feature visualization and exploration, and export of prioritization preferences and annotations (Figure 3.2). Below, we step through each individual section in detail.

Configuration and data import

Generation of the neoantigen candidate input files requires preprocessing using pVACseq starting from patient samples’ variant information (supplied as a VCF file). pVACseq produces neoantigen candidates with numerous features to be considered during prioritization. Two of pVACseq’s output files, an aggregated candidate file (tsv format) and a metrics file (json format), serve as input files to pVACview. The aggregated candidate file contains a list of all variants with summary level information such as: best predicted neoantigen candidate and its prediction score, DNA/RNA depth and variant allele frequencies, gene and allele expression, etc.
(for further details please refer to the online documentation at pvactools.org). The metrics json file contains extensive additional transcript and peptide-level information that is needed for certain features of the pVACview application.
Figure 3.2. Overview of example workflow for prioritizing neoantigens using pVACview.

pVACview can be broken down into three main sections: upload, visualize/explore, and export. When exploring the neoantigen candidates, users are presented with three levels of information: variants, transcripts and peptides. This example workflow guides the user through critical questions that need to be considered when making their decisions regarding the different candidates. Each section is accompanied by the corresponding feature in the pVACview interface.

Users have the option to additionally include a tsv file supplementing candidate information from a different set or class of HLA alleles. This allows users to view basic median binding information of Class II results while looking at detailed Class I prediction results or vice versa. For users investigating a specific gene set of their own interest, we provide the option of uploading a tsv file where each line contains an individual gene name. These genes, if found in the aggregate report file, will be highlighted in a green box with bold font in the Gene report column of the visualization interface.

Neoantigen visualization and exploration

Uploaded neoantigen candidates can be explored and analyzed in several different ways. Users are provided with neoantigen features that can be separated into three main categories: variant, transcript, and peptide.

Variant-level information is presented in the main aggregate report table, showcasing the best neoantigen candidates for each variant as well as genomic information, expression level, DNA and RNA variant allele frequency, average binding prediction scores, and overall summary of passing transcript and peptides. Users are additionally provided with a mutation and gene info box, which provides further information on the exact genomic location and nucleic acid mutation. We have also included a link to the OpenCRAVAT variant report for the respective variant\textsuperscript{208}. This report allows users to explore rich variant information including: variant effect
annotations, associated cancer types, population allele frequencies, clinical relevance, gene annotation, pathogenicity predictions, etc.

Once a specific variant is selected, users are provided with individual transcripts containing the variant. The selected variant may occur within multiple transcripts that may result in distinct neoantigen peptide sequences and those that produce good binding predictions against at least one HLA allele are shown in the transcript table. The expression level of each transcript is provided as further guidance when selecting the best neoantigen candidate. By selecting a specific transcript, users are then provided with a peptide table. The peptide table displays all peptide sequences from your selected transcript that were predicted to be good binders (for at least one HLA allele). Both mutant (MT) and wildtype (WT) sequences are shown, along with median binding affinities (where the MT score passed the binding threshold).

Users are also provided with binding and anchor information for each individual peptide. By selecting each pair of MT/WT peptides, users can access: 1) plots of the individual IC50-based binding affinity predictions of the MT and WT peptides for HLA alleles with at least one predicted strong binder 2) plots of the individual percentile-based binding affinity predictions and 3) a binding affinity table with numerical IC50 and percentile values. Note that these peptides each have up to 8 binding algorithm scores for Class I alleles (with pVACseq version 3.0) or up to 4 algorithm scores for Class II alleles. For each peptide table, we also provide users with an allele-specific anchor prediction heatmap, based on computational predictions from our previous work. These predictions are normalized probabilities representing the likelihood of each position of the peptide to participate in anchoring to the HLA allele. The top 30 MT/WT peptide pairs from the peptide table are shown with anchor probabilities overlaid as a heatmap. The anchor probabilities shown are both allele and peptide length specific. In the anchor heatmap
view, the mutated amino acid is marked in red (for missense mutations) and each MT/WT pair is separated from others using a dotted line.

After consulting the breadth of information displayed in pVACview, users can label each variant with their corresponding evaluation using the dropdown menu in the Eval column of the aggregate report. The number of evaluations performed (accept, reject, review) are tracked in the peptide evaluation overview section.

**Export of neoantigen evaluations and final report**

When users have either finished ranking neoantigen candidates or need to pause and would like to save current evaluations, they can export the current main aggregate report using the export page. We provide two download file types (tsv and excel). The excel format is user-friendly for downstream visualization and manipulation. However, if the user plans to continue editing the aggregate report and would like to load it back in pVACview with the previous evaluations preloaded, they must use the tsv format. This serves as a way to save progress as all evaluations are cleared upon closing or refreshing the pVACview app.

**3.4 Results and Discussion**

Multiple tools and workflows have been created for neoantigen characterization for studies of tumor T cell biology and the development of neoantigen-based therapies. Some existing tools that address individual factors for prediction of neoantigens do include visualization components (such as netMHC for binding predictions). However, pVACview is the first tool to present this volume of data together in a simple, integrated view. Previously, pipelines (e.g. pVACtools) included the algorithms for detecting, characterizing and prioritizing neoantigens from various sources. However, existing tools describe no aspect of visually exploring the results but rather produce a static neoantigen report often attempting to provide a
“best” neoantigen for each variant based on simple criteria such as binding affinity predictions. During this process, the field often over-simplifies the outputs to make them tractable, and thus results are fraught with assumptions about what “best” means. This reduces the ability to effectively prioritize neoantigen candidates and a more nuanced approach that allows the use of many more contextually relevant features would be more valuable. We therefore believe that an interactive neoantigen visualization tool, customized to this specific application is very much needed. As a dynamic interactive visualization interface, pVACview overcomes many limitations of tabular reports, allowing the user to consider neoantigens in the context of transcript expression, tumor clonality, multiple registers (peptides of the same length where the mutation is at different positions), peptide lengths, alternative transcript isoforms, an ensemble of predictive binding algorithms, HLA specific anchor information and much more.

While there has been a rapid development of sequencing technologies, bulk tumor tissue samples undergoing sequencing are often subject to purity issues. Additionally, intratumoral heterogeneity presents a considerable challenge to cancer therapies, making it critical to identify variants from the founding clone rather than those that are subclonal. Neoantigens arise from tumor-specific genomic variations and each variant can have multiple transcripts encompassing the variant location. Thus, not only should transcript-specific expression level be taken into account, transcripts resulting from different splicing patterns also may have considerable impact on the exact neoantigen peptide sequence. Once the correct sequence surrounding a variant from an expressed transcript is identified, neoantigen candidates can be extracted by looking at different registers and different lengths of peptides containing the amino acid mutation. This aspect is particularly important for cases where neoantigens are derived from frameshift variants. Frameshift variants can produce a large number of neoantigen candidates that vary in sequence
and mutation position, with each peptide having a different set of prediction scores. Only by looking across all corresponding candidates can the user make the most optimal decision for the variant of interest. The detailed information provided by pVACview can effectively help users address all these specific complexities.

Furthermore, pVACview’s drill down approach to information display helps researchers integrate variant clonality, Class I and II binding predictions, competing binding prediction algorithms, binding metrics and mutation positional information for each candidate neoantigen peptide. Expressed neoantigens of different lengths and registers from a founding clone of the tumor can potentially bind (or not bind) to either Class I or Class II HLA alleles, either through endogenous or exogenous pathways respectively on either the tumor cell or an antigen presenting cell. Each patient can have up to 6 different Class I HLA alleles, and up to 8 different Class II alleles (and dimer combinations of these). When evaluating neoantigen candidates in terms of binding predictions, researchers should not only take into account how well neoantigens are potentially binding to each patient-specific HLA allele, they should also consider how many different HLA alleles it can bind robustly. For each neoantigen, there are numerous prediction algorithm results available. How the scores are distributed across different algorithms and whether the IC50 binding prediction or percentile rank value should be utilized are all important aspects that require careful evaluation. If a neoantigen is expressed and predicted to bind well to an HLA allele, researchers should further consider the anchor and mutation locations of the peptide-MHC pairing. A subset of peptide positions are primarily presented to the T-cell receptor for recognition, while others are primarily responsible for anchoring to the MHC (though neither role is exclusive). Whether the mutation lies in an anchor region and how well the WT peptide
binds MHC creates different scenarios that can influence whether a neoantigen remains a good candidate.

Finally, in the setting of clinical trials, additional details influence the quality of neoantigens. Tumor samples are first collected from patients and put through a series of genomic analysis pipelines, including DNA and RNA sequencing, variant detection and expression estimation. Pipelines such as pVACtools then take these results and identify and characterize possible neoantigen candidates. Throughout this process, problems such as low tumor purity, contamination, and insufficient or excessive neoantigen candidates may arise. pVACview allows users to promptly adjust tiering of candidates based on tumor purity and expression levels. It also highlights the specific failing criteria, providing users the option to further explore criteria such as how to define anchor positions, with the flexibility of rescuing candidates for samples with insufficient candidates for downstream applications such as neoantigen vaccines. For cases where an excessive amount of neoantigen candidates exist, pVACview effectively prioritizes candidates (based on calculated tiering, allele expression and average mutant peptide binding affinity) while simultaneously allowing users to sort and annotate candidates based on features of their own choosing.

All the aspects described above are potentially critical in order to infer whether the presenting peptide sequence can successfully induce an immune response. pVACview was designed to present this complex information to researchers and aid in the prioritization and selection of neoantigen candidates for personalized cancer vaccines or other therapeutic and research applications. Researchers and clinicians can use pVACview to visualize neoantigen candidates along with detailed supporting information including that of the genomic variant,
transcripts affected by the variant, and good-binding peptides predicted from the respective transcripts.

3.5 Conclusions

Accurate neoantigen prediction is critical to cancer immunotherapy treatments and several tools have been built to account for individual aspects throughout this process. However, these tools lack methods for integration and visualization, making it challenging for researchers and clinicians to efficiently explore the many molecular and algorithmic features relevant to neoantigens. For example, variant, transcript and peptide level information are all critical in prioritizing neoantigen candidates for downstream applications. pVACview integrates multiple levels of information into a visualization tool, allowing users to analyze each candidate in detail for optimal decision making. This tool has been tested and used in clinical trials and research projects involving human, mouse, and canine model systems. We hope by using pVACview researchers can analyze and prioritize neoantigen candidates with greater efficiency and accuracy. The application is available as part of the pVACtools pipeline and as an online web tool at [www.pvacview.org](http://www.pvacview.org).

3.6 Software availability

*Data availability*

The FASTQ and BAM files for HCC1395 and HCC1395BL demonstration data can be found at NCBI BioProject accession number: PRJNA201238 (SRA accession number: SRX285805, SRX285804, SRX278523, SRX278522, SRX278521, SRX278520, SRX278519, SRX278518, SRX278517).

*Software availability*
The pVACview codebase is hosted publicly on GitHub at
https://github.com/griffithlab/pVACtools. User documentation is available at pvactools.org. This
project is licensed under the BSD 3-Clause Clear License
(https://github.com/griffithlab/pVACtools/blob/master/LICENSE). pVACtools has been
packaged and uploaded to PyPI under the “pvactools” package name and can be installed on
Linux systems by running the `pip install pvactools` command. Installation requires a Python 3.5
environment which can be emulated by using Conda. Versioned Docker images including all
dependencies are available on DockerHub (https://hub.docker.com/r/griffithlab/pvactools/).
Releases are also made available on GitHub (https://github.com/griffithlab/pVACtools/releases).

3.7 Acknowledgements

We thank the patients and their families for the donation of their samples for research purposes.
Malachi Griffith was supported by the National Human Genome Research Institute (NHGRI) of
the National Institutes of Health (NIH) under Award Number R00HG007940. Malachi Griffith
and Obi Griffith were supported by the NIH National Cancer Institute (NCI) under Award
Numbers U01CA209936, U01CA231844 and U24CA237719. Malachi Griffith was supported
by the V Foundation for Cancer Research under Award Number V2018-007. Malachi Griffith,
Obi Griffith, Huiming Xia, and William Gillanders were supported by the NCI under Award
Number U01CA248235.

3.8 Author Contributions

H.X. and M.G. contributed to the design of the model. H.X., S.K. and J.J.S. contributed
to writing the software. Z.L.S and B.F. contributed to generation of sample data. H.X., J.H.,
O.L.G. contributed to the overall supervision of the project. H.X. and J.M. wrote the manuscript and created all figures in consultation with M.G. and O.L.G. All authors provided critical feedback and helped shape the research, analysis and manuscript.
Chapter 4: Generating a standardized data set of neoantigen features using experimentally validated neoantigens collected from publications

4.1 Preamble
The work described in the chapter is an accumulation of effort from multiple people over many years during their time in the Griffith Lab, including: Jasreet Hundal, Yang-Yang Feng, Connor Liu, Jonathan Song, and myself. Jasreet, Yang-Yang and Connor collected raw sequencing data from published neoantigen studies over the past decade and started the analysis on the datasets. I then subsequently took the datasets and ran a CWL-based immunogenomic pipeline to generate a standardized set of neoantigen features. Jonathan was responsible for generating neoantigen features relating to subcellular compartments of the gene where the neoantigens were derived from as well as biophysical and biochemical properties.

4.2 Introduction
As discussed in Chapter 1, numerous factors are involved in the process of antigen generation, processing, binding and recognition, with tools built for each respective category. However, for a neoantigen candidate to be included into a personalized vaccine treatment for a cancer patient, all the factors above will need to be taken into account by the clinician or researcher reviewing the list of candidates. Thus, a number of bioinformatic pipelines have emerged with the goal of assembling available tools to streamline the neoantigen identification process. These pipelines attempt to address multiple factors that should be given careful consideration when predicting neoantigens for effective cancer treatments. These considerations include: use of multiple binding prediction algorithms (binding prediction variability/consensus), integration of DNA and
RNA data (neoantigen candidate genes/transcript expression and variant allele expression), variant phasing (proximal variants) \(^{32,117}\), interpretation of variants in the context of clonality/heterogeneity \(^{118}\), patient tumor HLA expression and somatic mutation, and prediction of tumor immunogenicity \(^{119,120}\).

Current neoantigen pipelines, such as pVACtools, assemble DNA and RNA sequencing data and present the variant allele frequencies and gene/transcript expression values of mutations for evaluation. After generating a list of neoantigens, certain tools also calculate the probability of cleavage, the location of cleavage as well as the TAP transport efficiency of each candidate \(^{209}\). Binding affinities of peptides to the patient-specific MHC complexes are subsequently predicted by calling available MHC binding prediction algorithms. While tools generate multiple peptide features for the user to navigate through, the implementation for recommending top neoantigen candidates is often a rather simple weighting of a small number of features. Example features include binding affinity, agretpoicity, expression and DNA/RNA VAF etc. However, pipelines, such as pVACtools, outputs additional data such as mutation position and peptide length but has yet to utilize them for prioritization. This is mainly due to our lack of knowledge regarding how each individual component could influence the effectiveness of a neoantigen candidate and how weights should be assigned to integrate them in creating an overall ranking system.

Previously, Kim et al. have published Neopepsee, a model that attempted to capture the contributions of nine immunogenicity features through training of traditional machine-learning based classifiers \(^{125}\), including random forest and Naive Bayes models. Additionally, researchers at Gritstone have also published an artificial neural network model, called EDGE, considering simple features, such as peptide sequences and RNA expression values, and trained their model
using mass spectrometry data collected from HLA immunoprecipitation. While both are improvements to the previously proposed prediction models which mainly utilize *in vitro* binding affinity results, they are still lacking in terms of training data as well as model complexity and consequently the predicted peptides still have a low yield when validated with T-cell activation assays.

Thus, to improve our current neoantigen prioritization workflow, we want to explore how we can algorithmically improve our current prioritization pipelines. To achieve this goal, we ultimately want to train a artificial neural network with a list of comprehensive features on high quality experimentally validated neoantigens, particularly neoantigens with proven efficacy in inducing T cell responses in patients (e.g. IFN-γ ELISPOT, MHC multimer analysis and other clinical approaches). Through this, we would aim to reduce the number of false positive neoantigen predictions for a significantly more accurate neoantigen prediction model.

However, the accuracy of such models rely heavily on its training data and whether we can reliably generate neoantigen features that we believe play a role in predicting T cell responses. Published neoantigen studies over the past decade range widely in 1) the type of input data, 2) the analysis pipeline used and 3) neoantigen features the original authors considered when prioritizing respective neoantigens for validation. This results in the lack of a standardized neoantigen dataset with features generated from the same immunogenomics analysis pipeline with the same level of quality control and other applied filters. In this chapter, we describe our on-going efforts to create such a dataset, which can then consequently be used to train and test different machine learning models and improve on current neoantigen predictors.
4.3 Data collection

The Griffith Lab has been collecting whole exome sequencing and RNA sequencing data from published datasets within the past decade. We focused our search on published cancer vaccine studies that met each of these requirements: 1) human study that had performed matched tumor and normal sequencing, 2) generated a list of candidate peptides, and 3) empirically validated a neoantigen specific T-cell response. We have collected data from over 23 neoantigen studies corresponding to X patient samples. Overall, these studies validated X neoantigen candidates, with X positives and X negatives (Table 4.1).

<table>
<thead>
<tr>
<th>Study Identifier</th>
<th>Study Description</th>
<th>Peptide Lengths</th>
<th>Validated Peptides</th>
<th>Validation Experiments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tran_PMID26516200</td>
<td>11 Dec 2015, GI cancer</td>
<td>25</td>
<td>412 tested, 16 positive</td>
<td>TMG + ELISpot</td>
</tr>
<tr>
<td>Cohen_PMID26389673</td>
<td>2015 Sep 21, metastatic melanoma</td>
<td>8-11</td>
<td>10 tested, 10 positive</td>
<td>TMG + ELISpot</td>
</tr>
<tr>
<td>Gros_PMID26901407</td>
<td>2016 Feb 22, melanoma</td>
<td>8-11</td>
<td>575 tested, 6 positive</td>
<td>TMG + ELISpot</td>
</tr>
<tr>
<td>Tran2_PMID24812403</td>
<td>2019 Aug 8, epithelial cancer</td>
<td>25</td>
<td>25 tested, 1 positive</td>
<td>TMG + ELISpot</td>
</tr>
<tr>
<td>Rajasagi_PMID24891321</td>
<td>2014 June 2, leukemia</td>
<td>9-10</td>
<td>42 tested, 3 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Linnemann_PMID25531942</td>
<td>2014 Dec 22, melanoma</td>
<td>31</td>
<td>8 tested, 8 positive</td>
<td>Flow cytometry</td>
</tr>
<tr>
<td>Stronen_PMID27198675</td>
<td>December 14, 2016, melanoma</td>
<td>9-10</td>
<td>54 tested, 11 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Wick_PMID24323902</td>
<td>2014 Mar 1, ovarian cancer</td>
<td>15</td>
<td>68 tested, 1 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Pritchard_PMID26048577</td>
<td>2015 June 5, melanoma</td>
<td>21</td>
<td>612 tested, 5 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Nielsen_PMID26631611</td>
<td>2015 Dec 2, lymphoma</td>
<td>9-11</td>
<td>15 tested, 3 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Robbins_PMID23644516</td>
<td>2013 Dec 1, melanoma</td>
<td>9-10</td>
<td>221 tested, 8 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Lu_PMID24987109</td>
<td>2015 Jul 1, melanoma</td>
<td>25</td>
<td>288 tested, 2 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Kosaloglu_PMID27999735</td>
<td>2016 Jul 22, neuroendocrine</td>
<td>8-11</td>
<td>28 tested, 4 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Snyder_PMID25409260</td>
<td>2014 Nov 19, melanoma</td>
<td>9</td>
<td>2 tested, 2 positive</td>
<td>ELISpot</td>
</tr>
<tr>
<td>Prickett_PMID27312342</td>
<td>2016 Jun 16, melanoma</td>
<td>12</td>
<td>720 tested, 10 positive</td>
<td>TMG + ELISA</td>
</tr>
<tr>
<td>Bassani-Sternberg_PMID27869121</td>
<td>2016 Nov 21, melanoma</td>
<td>9</td>
<td>8 tested, 2 positive</td>
<td>ELISpot</td>
</tr>
</tbody>
</table>
Table 4.1 Summary of neoantigen related studies collected with T-cell validation datasets.

For each neoantigen candidate, we generated a number of features prior to obtaining full genomic analysis results, including: hydrophobicity, net charge, peptide size, polarity as well as subcellular compartments of the neoantigen-originating protein in terms of target destination.

4.4 NGS sequencing analysis

All raw sequencing files underwent the same analysis pipeline which includes: 1) RNA-seq analysis, 2) somatic variant analysis, 3) germline variant analysis, 4) phasing and 5) neoantigen identification. Important steps involved in these processes are outlined as follows.

RNA-seq analysis begins with adapter trimming of raw sequence data using Flexbar and subsequent alignment using STAR aligner to the reference genome (build GRCh38). Gene and transcript expression levels are obtained using Kalliso and Stringtie in reference guided modes. QC metrics are obtained using tools including FastQC, Picard CollectRnaSeqMetrics, RSeqQC and multiQC as the overall summarization tool. Somatic variant analysis starts with aligning
DNA sequence reads to the reference genome using BWA-mem aligner. Alignments are subject to base quality score recalibration and QC metrics are obtained using Picard and samtools flagstat. Missense variants and small insertions and deletions are called using a variety of detection tools including Varscan, Strelka, MuTect and Pindel. Somatic variants from these callers are standardized using GATK LefAlignAndTrim and combined into a single VCF file using GATK CombineVariants with multiallelic sites separated using vt decompose. High confidence somatic variants are annotated with VEP for transcript variant effect with additional annotations from COSMIC and ClinVar. Variant allele frequencies are calculated using bam-readcount (https://github.com/genome/bam-readcount). For germline analysis, DNA samples of the patient’s normal samples are first used as input to OptiType69, for HLA Class I typing, and PHLAT71, for Class II typing. These are then cross-referenced and compared with reports from the original studies when available. Germline variant analysis is performed using GATK HaplotypeCaller and subsequently annotated using VEP as well. In the phasing step, somatic variants with germline variants in close proximity (and in phase) are both carefully accounted for when generating the correct neoantigen candidate sequences. In order to achieve this, phased proximal variants are incorporated into the somatic VCF file. VAtools (https://github.com/griffithlab/VAtools) is used for subsequent readcount, expression and genotype annotation. The fully annotated VCF is supplied as input to pVACtools164 for neoantigen identification and prioritization. Peptide candidates generated from each somatic mutation undergo binding affinity predictions using up to eight algorithms for Class I and up to four for Class II. Multiple neoantigen features are also generated and summarized by pVACtools including: 1) best binding peptide and HLA pairing, 2) gene and transcript expression, 3)
mutation position and anchor locations, 4) predictions based on elution/presentation data, 5) genomic information such as DNA/RNA VAF and RNA depth.

4.5 Results
Configuring and running the full immunogenomics pipeline on the samples we have collected is a timely process that is still ongoing. To date, we have fully analyzed 5 out of 23 collected studies, corresponding to 24 patient samples. With this initial dataset, we are able to start performing some feature characterization using these preliminary results obtained. Positive and negative validation outcome groups showed a significant difference in distribution when comparing median IC50, median percentile, gene expression as well as transcript expression. Positive peptides in general have a lower binding prediction (median and percentile) as well as higher gene and transcript expression (Figure 4.1).
**Figure 4.1. Comparison of individual features between positive and negative validation dataset.** Validated neoantigen candidates were split based on their validation outcome and compared for four different features, including median IC50 binding prediction (nM), median percentile binding prediction (%), gene expression (tpm) and transcript expression (tpm). Negative outcome group is shown in blue and the positive outcome group is shown in green.

Most algorithms utilized by pVACtools (with the exception of MHCnuggets) provide both IC50 as well as percentile binding predictions. When comparing between the two different validation outcome groups broken down by individual algorithms, best MT percentiles were observed to better distinguish the two groups apart, whereas best MT IC50 scores between positive and negative peptides were on relatively similar levels, especially for algorithms SMM and SMMPMBEC (Figure 4.2, Table 4.2).

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Median IC50 (nM) binding (positive)</th>
<th>Median IC50 (nM) binding (negative)</th>
<th>Difference in median IC50</th>
<th>Median Percentile (%) (positive)</th>
<th>Median Percentile (%) (negative)</th>
<th>Difference in median percentile</th>
</tr>
</thead>
<tbody>
<tr>
<td>MHCflurry</td>
<td>4557</td>
<td>9906</td>
<td>5349</td>
<td>6.37</td>
<td>14.14</td>
<td>7.77</td>
</tr>
<tr>
<td>MHCnuggetsI</td>
<td>954</td>
<td>3766</td>
<td>2812</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>NetMHC</td>
<td>5922</td>
<td>16726</td>
<td>10804</td>
<td>4.59</td>
<td>16.65</td>
<td>12.06</td>
</tr>
<tr>
<td>NetMHCcons</td>
<td>3750</td>
<td>13199</td>
<td>9449</td>
<td>6.58</td>
<td>25.48</td>
<td>18.90</td>
</tr>
<tr>
<td>NetMHCpan</td>
<td>5720</td>
<td>17026</td>
<td>11306</td>
<td>5.60</td>
<td>20.44</td>
<td>14.84</td>
</tr>
<tr>
<td>PickPocket</td>
<td>4402</td>
<td>10628</td>
<td>6224</td>
<td>5.11</td>
<td>18.53</td>
<td>13.42</td>
</tr>
<tr>
<td>SMM</td>
<td>373</td>
<td>593</td>
<td>220</td>
<td>1.19</td>
<td>9.88</td>
<td>8.69</td>
</tr>
<tr>
<td>SMMPMBEC</td>
<td>324</td>
<td>786</td>
<td>462</td>
<td>1.44</td>
<td>9.36</td>
<td>7.92</td>
</tr>
</tbody>
</table>

Table 4.2 Comparison across different algorithm predictions between positive and negative validation outcome datasets
Figure 4.2 Comparison of predicted IC50 binding affinity and percentile values across different algorithms and validation outcome groups. Top: IC50 binding affinity values are plotted across all available class I algorithms in pVACtools. Bottom: Percentile values plotted for all seven algorithms where available (MHCnuggetsI did not provide such values). Y-axis is in log scale for both top and bottom plots. Positive validation outcome group is in green and the negative outcome group is in blue.
While these preliminary analyses provided interesting insights, we also encountered various issues including loss of neoantigen candidates with validation outcome due to the absence of certain variants. Specifically, when comparing the variants detected through our pipeline with those described in the corresponding publications, we observed that 9 out of 51 total positive and 740 out of 2103 total negative neoantigens were not detected by our pipeline (Table 4.3).

<table>
<thead>
<tr>
<th>Paper ID</th>
<th>Patient ID</th>
<th>Positively Validated Neoantigens</th>
<th>Negatively Validated Neoantigens</th>
<th>Published positives not detected in our pipeline</th>
<th>Published negatives not detected in our pipeline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3812</td>
<td>0</td>
<td>169</td>
<td>0</td>
<td>125</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3942</td>
<td>3</td>
<td>138</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3948</td>
<td>2</td>
<td>168</td>
<td>0</td>
<td>107</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3971</td>
<td>0</td>
<td>109</td>
<td>0</td>
<td>55</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3978</td>
<td>1</td>
<td>32</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt3995</td>
<td>2</td>
<td>122</td>
<td>0</td>
<td>72</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt4007</td>
<td>2</td>
<td>215</td>
<td>0</td>
<td>93</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt4032</td>
<td>3</td>
<td>187</td>
<td>0</td>
<td>116</td>
</tr>
<tr>
<td>Tran_PMID26516200</td>
<td>pt4069</td>
<td>0</td>
<td>79</td>
<td>0</td>
<td>68</td>
</tr>
<tr>
<td>Cohen_PMID26389673</td>
<td>pt3466</td>
<td>4</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Cohen_PMID26389673</td>
<td>pt3703</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cohen_PMID26389673</td>
<td>pt3713</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cohen_PMID26389673</td>
<td>pt3919</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Gros_PMID26901407</td>
<td>pt3784</td>
<td>5</td>
<td>143</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>Gros_PMID26901407</td>
<td>pt3903</td>
<td>6</td>
<td>309</td>
<td>0</td>
<td>22</td>
</tr>
<tr>
<td>Gros_PMID26901407</td>
<td>pt3926</td>
<td>0</td>
<td>107</td>
<td>0</td>
<td>22</td>
</tr>
<tr>
<td>Gros_PMID26901407</td>
<td>pt3998</td>
<td>10</td>
<td>104</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>Robbins_PMID23644516</td>
<td>pt2369</td>
<td>3</td>
<td>98</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>Robbins_PMID23644516</td>
<td>pt2098</td>
<td>2</td>
<td>49</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>Robbins_PMID23644516</td>
<td>pt3309</td>
<td>3</td>
<td>74</td>
<td>0</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Total number of positives detected</th>
<th>Total number of negatives detected</th>
<th>Remaining positives</th>
<th>Remaining negatives</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>42</td>
<td>1363</td>
<td>9</td>
<td>740</td>
</tr>
</tbody>
</table>

Table 4.3 Summary of neoantigen candidates produced from in-house immunogenomics pipeline compared to published candidates
Investigation into corresponding variants of where the missing neoantigens originated showed multiple reasons why variants may be absent, including: low VAF, low number of read counts, low mapping quality, low base quality, and even difference in gene annotations between versions used in the original study and that of our in-house pipeline. Since our established immunogenomics pipeline contains multiple filtering steps regarding variant quality, these cases will have to be manually reviewed for further action.

4.6 Discussion

While previously researchers have made various efforts to utilize machine learning approaches for neoantigen prediction and prioritization, training dataset for these algorithms consisted of either in vitro binding data or in vivo elution output or a combination of the two. However, binding and presentation does not directly imply a T-cell response to the given neoantigen candidate. Thus current models remain suboptimal and improvement is needed considering the low yield of positive neoantigens when validated with T-cell activation assays. The Griffith lab has been collecting whole exome sequencing and RNA sequencing data from published datasets. By running raw sequencing data through our established immunogenomics pipeline, we hope to generate a standardized neoantigen feature dataset with validation results, able to be used in supervised ML training models. Preliminary analysis of pipeline results show promising correlations when considering single features such as median MT IC50, median MT percentile, gene expression and transcript expression. However, there also exists a significant difference between somatic variants detected from our established immunogenomics pipeline and the variants reported and tested in the original publications. To address this, we could create our own VCF representation directly from variants reported in publications. By using TransVar we can
obtain genomic coordinates for each variant given protein-level information. This custom VCF can then be annotated using bam-readcount to produce various metrics hinting at the quality of the variant. Information generated can be used to perform an ad hoc analysis to see if distinct patterns exist for variants removed by our analysis pipeline. Further investigation is needed to evaluate whether removing variants of low quality can influence the difference between positive and negative outcome groups for various neoantigen features.
Chapter 5: Conclusions and Future Directions

Neoantigen targeting therapies including personalized vaccines have shown promise in the treatment of cancers, particularly when used in combination with checkpoint blockade therapy. At least 60 clinical trials involving these therapies are underway across the world (neoantigen.clinwiki.org). Accurate identification and prioritization of neoantigens is highly relevant to designing these trials, predicting treatment response, and understanding mechanisms of resistance. In this research, we describe our efforts to improve upon current neoantigen identification and prioritization methods from various aspects.

5.1 Conclusions and future directions for prediction of allele-specific anchors

Neoantigen identification and prioritization relies on correctly predicting whether the presenting peptide sequence can successfully induce an immune response. As the majority of somatic mutations are single nucleotide variants, changes between wildtype and mutated peptides are typically subtle and require cautious interpretation. A potentially underappreciated variable in neoantigen-prediction pipelines is the mutation position within the peptide relative to its anchor positions for the patient’s specific MHC molecules. While a subset of peptide positions are presented to the T-cell receptor for recognition, others are responsible for anchoring to the MHC, making these positional considerations critical for predicting T-cell responses.

The objective of the anchor project was to predict allele-specific MHC anchor locations and investigate whether use of this information could significantly influence neoantigen identification and prioritization. We established a computational approach based on an ensemble of MHC binding prediction algorithms and exhaustive in silico mutation of peptides. By application of this approach to 609,807 distinct peptide sequences we were able to predict anchor
locations for 328 common HLA alleles with peptides of varying lengths (8-11 mers). Anchor location scores were clustered using hierarchical clustering and representative trends were identified and experimentally validated using various methods, including: x-ray crystallography structures, IC50 competition binding assays, and cell-based stabilization assays. All stabilization assays were completed 2-3 times in duplicate (N=4-6). Use of allele-specific anchor predictions was compared against conventional prioritization methods to demonstrate the potential influence on neoantigen prioritization results.

*Future directions*

As stated in our original manuscript, there are multiple future directions this study could improve upon. First and foremost, while our anchor results represent overall averaged scores across multiple neoantigens, slight variations of patterns do exist among different peptides for the same HLA allele. While it is currently computationally prohibitive to perform our workflow on a large scale for all possible neoantigen candidates, as a compromise, one could repeat our detailed process to generate peptide specific anchor predictions after performing other filtering strategies and arriving at a shortlist of candidates. Secondly, our work could be expanded to a wider range of HLA alleles. By starting with the wildtype proteome rather than cancer-specific mutations, a larger HLA-peptide seed dataset could be achieved through a wide-scale prediction of strong binders for rarer HLA alleles. With additional data generation and further research, we can also look into the expansion of our workflow to address Class II anchors and its influence on neoantigen prioritization. Furthermore, while we used various methods to validate our anchor location predictions, ideal experimental validation would involve mutating neoantigens designed to induce T-cell activation to explicitly showcase the importance of our results in clinical settings. Last but not least, anchor location scores may serve as an additional feature in machine
learning model training and testing. This will allow for a more nuanced approach where anchor scores may be weighted for each allele-peptide pairing accordingly in ML models.

5.2 Conclusions and future directions for pVACview

Numerous factors must be considered when prioritizing neoantigens for use in personalized therapies. Complexities such as alternative transcript annotations, multiple algorithm prediction scores and variable peptide lengths/registers all potentially impact the neoantigen selection process. While computational pipelines such as pVACtools generate numerous algorithmic predictions for neoantigen characterization, results from these pipelines are difficult to navigate and require extensive knowledge of the underlying tools for accurate interpretation. This often leads to over-simplification of pipeline outputs to make them tractable.

Due to the intricate nature of such neoantigen features, presenting all relevant information to experimentalists and immunogenomics tumor boards for candidate selection is a difficult challenge that current tools fail to resolve. In response, we have created pVACview, the first interactive tool designed to aid in the prioritization and selection of neoantigen candidates for personalized neoantigen therapies including cancer vaccines. pVACview has a user-friendly and intuitive interface where users can upload, explore, select and export their neoantigen candidates. The tool allows users to visualize candidates across three different levels, including variant, transcript and peptide information. We expect pVACview will allow researchers and clinicians to analyze and prioritize neoantigen candidates with greater efficiency and accuracy. The application is available as part of the pVACtools pipeline at pvactools.org and an online server can be accessed at pvacview.org.

Future Directions
We recognize that pVACview, in its current form, is restricted to use of input files generated from pVACseq, which limits its application to a broader audience. To address this, we have performed an extensive survey of currently available neoantigen prediction and prioritization pipelines. However, our survey showed that a significant number of tools are no longer actively being supported nor updated with newer features. Additionally, only one tool (NeoPredViz of NeoPredPipe\textsuperscript{118}) has a visualization component, with the features offered by the tool being extremely limited. The utility of a visualization software only goes as far as what is in the underlying data that is provided in the input files and thus we further analyzed a subset of popular neoantigen pipelines using the HCC1395 dataset, including NeoFox\textsuperscript{203}, VaxRank\textsuperscript{9}, antigen.garish\textsuperscript{213} and NeoPredPipe\textsuperscript{118}. By comparing the outputs of the surveyed pipelines, we observed a lack of data on multiple levels, all of which would be not only required for pVACview to reach its full functionality but also for users to have a comprehensive view of the neoantigen landscape.

However, we also acknowledge that out of the tools we examined, NeoFox used a variety of neoantigen feature generating tools that are not currently being utilized by pVACtools. We saw great value in providing pVACview users with NeoFox’s feature annotations. Thus, we are in the process of building an additional module that takes NeoFox results (tsv file with > 90 columns) as input and adding various visualizations to allow for user exploration. We also plan to make available a custom script to convert pVACtools output into the required input format for NeoFox with corresponding test datasets. We understand the broader research community may use a range of tools in their own neoantigen prediction process. Thus, in the future we also aim to add a custom module supporting tsv input files where users can specify which neoantigen feature
they would like to group their results by. These grouped neoantigens can then be visualized in various ways.

As neoantigen-related features continue to be added to computational pipelines, they will be actively incorporated into pVACview. By providing all related information in an intuitive layout, we hope to enable clinicians and researchers to make effective and accurate prioritization decisions on neoantigens for cancer immunotherapies.

5.3 Future directions for algorithmic improvement of neoantigen prioritization

Machine learning algorithms have been widely applied in the context of neoantigen binding predictions. While there has been previous efforts showing improved neoantigen prediction and prioritization models by utilizing machine learning approaches trained on in vivo binding affinity results, current models remain suboptimal in terms of training data as well as model complexity resulting in a low yield of positive neoantigens when validated with T-cell activation assays. Our lab has been collecting over the past years, whole exome sequencing and RNA sequencing data from published datasets. By running raw sequencing data through our established immunogenomics pipeline, we are generating a standardized neoantigen feature dataset with validation results, able to be used in supervised ML training models.

Future directions

Upon generating a clean dataset with neoantigen features and validation results, we can then use this data to first conduct preliminary feature selection in order to achieve better classification accuracy and reduce the risk of overfitting, including common measures such as correlation-based feature selection, information gain-based feature selection, and classification power achieved by a single feature. In order to measure the classification power that can be achieved by a single feature, we can train classifiers for each feature using different learning
models (e.g. Gaussian naïve Bayes (GNB), locally weighted naïve Bayes (LNB), random forest (RF), and support vector machine (SVM)) based on the same training dataset. We can then calculate the area under a curve (AUC) of the Receiver Operating Characteristic (ROC) curve to evaluate the different learning models. Additionally, both Spearman and Pearson correlations can be used to calculate between each feature to measure interdependencies.

Using experimentally validated positive and negative neoantigen sequences with key features identified, we can then train and test machine learning models to algorithmically improve upon current neoantigen prediction tools. For example, we can utilize a combination of neural networks with a supervised learning algorithm. The model will mainly be composed of neural networks trained on features in two categories based on our biological understandings: allele-interacting and non allele-interacting features. Allele-interacting features are features that are HLA allele dependent. This includes features that may vary in importance and weighting depending on the patient-specific HLA allele. Peptide sequences, anchor positions, N & C-terminal residues and other peptide specific features all fall into this category. However, other features such as peptide stability, TAP transport efficiency and protein cleavage scores contain information critical to the probability of peptide presentation but are not conveyed to the HLA molecule. Peptide sequences can be vectorized using a one-hot coding scheme and represented as fixed-length vectors by adding a padding character to the amino acid alphabet. Position specific features of the peptide, if considered after feature selection, will also be encoded using binary vectors. Evaluation of the model can be done using a 10-fold cross validation method on the constructed dataset. However, for the complex neural net that we are constructing, lack of sufficient training data may become a key concern. Potential solutions could involve: 1) adding peptides that have been shown to present on MHC molecules using mass spectrometry
approaches, 2) continue data collection and neoantigen feature generation with results coming from ongoing clinical trials, 3) switching to traditional machine learning algorithms such as Random Forest, which have previously shown to have moderate performance when applied to \textit{in vitro} validation datasets. For performance evaluation, the model can then be compared to other existing models including pVACtools\textsuperscript{8}, MHCflurry\textsuperscript{102} as well as Neopepsee\textsuperscript{125}.

Furthermore, we can also look into generating additional features with our collected dataset to account for important aspects currently overlooked. Examples include: 1) mutations of HLA alleles and cofactors that govern a proper T-cell response and subsequent cell-death, 2) the expression levels of genes critical to MHC binding, proper T-cell recognition and subsequent cell death signaling pathways. Mutations in these genes can result in MHC Class I antigen presentation machinery loss of function and resistance to immunotherapy. These factors include B2M, PD-1/PD-L1, interferon-receptor–associated Janus kinase 1 (JAK1), Janus kinase 2 (JAK2), as well as other elements involved in cell death signaling pathways. Overall, once we generate a standardized dataset with a list of comprehensive features on experimentally validated neoantigens, we ultimately hope to train machine learning models such as artificial neural networks and allow clinicians to better predict and prioritize neoantigens to provide patients with more effective personalized cancer immunotherapy treatments.

5.4 Summary

Numerous factors are involved in the process of antigen generation, processing, binding and recognition and accurate computational approaches are critical for filtering and prioritizing neoantigens. We want to improve our current neoantigen prioritization workflow by 1) better utilizing features of neoantigens that are currently available from pipelines such as pVACtools and 2) algorithmically improve our current prioritization pipelines. To better utilize features
currently made available from pVACtools, we first computationally predicted anchor positions for common HLA alleles and investigated how anchor and mutation positions influence priority decisions for neoantigen candidates. The predicted anchor results, along with other neoantigen features produced by pVACtools, have been incorporated into pVACview, a visualization application that aims to increase the efficiency and accuracy of neoantigen prioritization. With the goal of further algorithmically improving our prioritization pipelines, we collected raw sequencing data from neoantigen studies with T-cell validation data and are in the process of generating a standardized set of neoantigen features. This will enable us to train machine learning models on high quality experimentally validated neoantigens, particularly neoantigens with proven efficacy in inducing T cell responses in patients. Through these efforts, we expect to significantly improve the accuracy and efficiency of neoantigen prioritization for future patient treatment involving personalized cancer vaccines.
Appendix : Chapter 1 Supplemental Tables

Table 1.1 Tool categories, brief description of their role and list of exemplar tools

A compilation of the current state of tools, databases and other resources used in neoantigen pipelines. While many steps outlined may involve integration of multiple tools for comparable predictions (e.g. using multiple somatic variant callers or MHC binding affinity predictors) this table summarizes more options than are needed in a single workflow. For an example of the specific combination of tools, parameter settings, and order of operations used in a real end-to-end workflow based on our own practices please refer to our online tutorial for precision medicine bioinformatics (https://pmbio.org/).

<table>
<thead>
<tr>
<th>Tool Categories</th>
<th>Function &amp; Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alignment</td>
<td>DNA: Bwa-mem 214</td>
</tr>
<tr>
<td></td>
<td>RNA: STAR 215, HISAT2 216</td>
</tr>
<tr>
<td>Variant Callers</td>
<td>SNV/Indel: Mutect 19, Strelka 20, VarScan2 21, SomaticSniper 22, Shimmer 218, VarDict 219, deepSNV 220, EBCall 40</td>
</tr>
<tr>
<td></td>
<td>Structural Variants: Pindel 43, Manta 221, Lumpy 222</td>
</tr>
<tr>
<td></td>
<td>Fusions: STAR-Fusion 48, Pizzly 47, SOAPfuse 223, JAFFA 49, ChimPipe 224, GFusion 50, INTEGRATE 51</td>
</tr>
<tr>
<td><strong>VCF manipulation</strong></td>
<td><strong>Variant annotation</strong></td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Gene/transcript abundance estimation</strong></th>
<th><strong>HLA typing</strong></th>
</tr>
</thead>
</table>

|----------------------------------------|------------------|

<table>
<thead>
<tr>
<th>Neoantigen Prioritization pipelines</th>
<th>Class II Predictors: SMMAlign, NNAlign, ProPred, NetMHCII(2.3) and NetMHCIIpan(3.2), TEPITOPE, TEPITOPEpan, RANKPEP, MultiRTA, OWA-PSSM.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peptide Creation &amp; Delivery</td>
<td>pVACtools (^8) (pVACvector), Vaxrank (^9) (manufacturability).</td>
</tr>
<tr>
<td>TCR repertoire Profiling</td>
<td>LymAnalyzer, MiXCR, MIGEC, pRESTO, TRUST, TraCeR, VDJtools, VDJviz, ImmunoSEQ, GLIPH.</td>
</tr>
<tr>
<td>Immune Cell Profiling</td>
<td>CIBERSORT, TIMER, quanTIseq, immunophenogram, MCPcounter, SSGSEA.</td>
</tr>
</tbody>
</table>

Table 1.2 Key analysis considerations and practical guidance for clinical neoantigen workflows

A detailed summary of analysis and interpretation best practices and nuances that should be considered in implementing a neoantigen identification workflow. Topics are covered in order corresponding to the flow of major steps discussed in the main body and depicted in Figure 1. For further nuanced details on how to put the following guidance into practice please refer to our tutorial for precision medicine bioinformatics (https://pmbio.org/).
<table>
<thead>
<tr>
<th>Analysis area</th>
<th>Guidance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference genome sequences</td>
<td>The choice of human reference genome sequences can have important implications for various analysis steps throughout neoantigen characterization workflows. A consistent build/assembly (e.g. GRCh38 or GRCh37) of the genome should be used throughout the analysis. Even if two resources provide annotations based on the same assembly they may organize or name sequences differently and follow different conventions for representing ambiguous or repetitive sequences. They may also drop some sequences (e.g. alternate contigs) or add sequences that are not part of the official assembly (e.g. “decoy” sequences). Using reference files from multiple sources for different tools is difficult to avoid but should be pursued cautiously. For example, the naming of chromosomes and contigs used for DNA read alignment and variant calling should be compatible (identical) to those used in transcript annotations. Otherwise this may prevent correct prediction of protein sequences for neoantigens.</td>
</tr>
<tr>
<td>Use of alternate contigs in the reference genome</td>
<td>The inclusion or exclusion of alternate contigs from the latest human reference genome build can have important implications for HLA typing tools such as xHLA (^74). In particular, if a tool assumes that all relevant reads for HLA typing can be extracted from an existing alignment (rather than performing de novo re-alignment of all reads), it matters whether some of these reads may have been placed on alternate contigs</td>
</tr>
</tbody>
</table>
for the HLA locus of chromosome 6. Some HLA typing approaches avoid this issue by aligning all reads directly to a database of known HLA gene sequences (e.g. from the IPD-IMGT/HLA resource). This has the disadvantage that without competitive alignment of each read to the whole genome, some reads may be misaligned to these sequences and may affect accuracy during HLA typing. A reference genome alignment approach where the diversity of HLA loci is properly represented in the reference avoids this concern and has the potential to leverage alignments that may have already been produced for variant calling. For example, all reads aligning to the HLA loci of chromosome 6, the corresponding alternate contigs (if present in the reference), and unaligned reads could be extracted from a BAM file and used for HLA typing.

<table>
<thead>
<tr>
<th>Transcript annotation build versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transcript annotation resources (e.g. Ensembl, RefSeq, GENODE, Havana) update their transcript sequences and associated annotations more frequently than the reference genome sequence build/assembly. For example, Ensembl is currently on version 96, the 21st update since the latest release of the human reference genome build GRCh38. As with reference genome builds, it is highly desirable to use a consistent set of transcript annotations across the steps of a neoantigen characterization workflow. For example, the same transcripts should be used to annotate somatic variants as those used to estimate transcript</td>
</tr>
</tbody>
</table>
and gene abundance from RNA data.

| Variant detection sensitivity | Correct neoantigen identification and prioritization rely on both somatic and germline variant detection (for proximal variant analysis) as well as variant expression analysis. QC analysis of both DNA and RNA data should be performed to assess the potential for a high false negative rate in detecting somatic variants that might lead to neoantigens, identify germline variants in phase with somatic variants that influence the peptide sequence bound by MHC, or to assess expression of these variants. Tumor samples vary significantly in their level of purity and genetic heterogeneity. To achieve high sensitivity for variant detection, common strategies involve increasing average sequencing depth and combining results from multiple variant callers. |
| Combining variants from multiple callers | The majority of somatic variant callers now use the widely adopted variant call format (VCF). Furthermore, many toolkits now exist for manipulation of these files, including merging. However, due to the complexity and flexibility of the VCF specification (https://samtools.github.io/hts-specs/VCFv4.2.pdf), the existence of multiple versions of the specification, and varying interpretations of VCF rules observed in the output of somatic variant callers, great care must be taken in combining multiple VCFs and using these merged results. Important considerations include: (A) Variant justification and parsimony such as left aligning or trimming variants to harmonize those |
that can be correctly represented at multiple positions without changing
the resulting sequence. (e.g. GATK LeftAlignAndTrimVariants) (B)
Normalization of multi-allelic variants by separating multiple variant
alleles that occur at a single position onto multiple lines in a VCF. (e.g.
vt decompose) (C) Harmonization of sequence depths, allele depth, and
allele fraction values that may be calculated inconsistently by different
variant callers through the use of an independent counting tool such as
bam-readcount (https://github.com/genome/bam-readcount). (D)
Determining the final status for each variant (PASS or filters failed).
(e.g. GATK SelectVariants) (E) Choosing the variant INFO and
FORMAT fields to represent in the final merged VCF.

| Variant refinement  | Somatic variant calling pipelines remain subject to high rates of false
| (“manual review”)   | positives, particularly in cases of low tumor purities, insufficient depth
|                    | of sequencing of tumor (or matched normal) samples or sub-clones.
|                    | Prior to final neoantigen selection, all somatic variants should be
carefully reviewed for possible alignment artifacts, systematic
sequencing errors, nearby in-phase proximal variants, and other issues
using a standard operating procedure for variant refinement such as that
outlined by Barnell et al. 27.

| Choosing RNA and DNA VAF cutoffs | It is impossible to define universal variant allele fraction (VAF)
|                                 | recommendations because of the varying distribution of VAFs observed
|                                 | for tumor samples with different sequencing depth, tumor |
purity/cellularity, genetic heterogeneity and degree of aneuploidy. In general, neoantigens corresponding to somatic variants with higher VAFs (in both DNA and RNA) will be considered with higher priority. However, the interpretation of each individual candidate may be influenced by one or more of these factors. Estimating the overall purity of the DNA sample by VAF distribution and distinguishing founding clones from sub-clones requires accurate assignment of each variant to a copy number estimate. Accepting or rejecting candidates on the basis of VAF requires a nuanced approach that takes the characteristic of each tumor into account. For example, a variant with a relatively low DNA VAF may be accepted in some cases if sequencing depth at the variant position was marginal, leading to a less accurate VAF estimate. A variant with a relatively high DNA VAF may be rejected if RNA-seq analysis shows strong evidence for allele specific expression (of the wild type allele).

| Interpretations that depend on RNA quality assessment | Attempting to define expressed and unexpressed variants by RNA-seq analysis is a common feature of many neoantigen characterization workflows. Applying hard filters in this area should be pursued with great caution. All interpretation of RNA-seq should be accompanied by comprehensive QC analysis of the data \(^{256}\). A lack of evidence for expression in RNA-seq data may not be definitive evidence for non-expression of a variant since not all genes can be robustly profiled by |
RNA-seq (e.g. very small genes may be poorly detected by standard RNA-seq libraries \(^{257}\)). Tumor samples obtained in clinical workflows, particularly those involving FFPE, may frequently result in poor quality RNA samples. In these cases, the requirements for expression support may be relaxed in nominating neoantigen candidates. Furthermore, some variants occur within a region of a gene that is difficult to align reads to. In these cases, robust apparent expression of the gene may still be used to nominate a neoantigen even in the absence of evidence supporting expression of the variant allele itself. Use of spike in control reagents and routine profiling of reference samples can be helpful in determining consistent expression value cutoffs (e.g. FPKM or TPM values) across samples. In the absence of reliable gene or variant expression readout for an individual tumor, robust expression of the gene in tumors of the same type may be used to prioritize neoantigens.

| Assessing variant clonality | A major consideration in the interpretation of DNA VAFs of variants is the assessment of tumor clonality. Neoantigens corresponding to variants residing in the founding clone are inherently more valuable therapeutically than those residing in tumor sub-clones because the former have the potential to target elimination of all tumor cells. In personalized cancer vaccine designs, after correcting for ploidy and tumor purity, VAFs should be interpreted to prioritize neoantigens that correspond to founding clones. |
| Variant types and agretopicity | Calculation of “agretopicity”, or “differential agretopicity index” \(^{121}\), or “wild type / mutant binding affinity fold change” refers to an attempt to estimate the degree to which a neoantigen’s ability to bind to MHC differs from its corresponding wild type sequence. This calculation thus depends on the ability to define a wild type counterpart for each neoantigen sequence. For non-synonymous SNVs, the wild type counterpart sequence is assumed to be a peptide of the same length without the amino acid substitution. For many other variant types, defining a counterpart wild type sequence is much less obvious because the variant may lead to a sequence that is entirely novel and bears little or no homology with the wild type sequences encoded from the region of the variant. These include frameshift mutations caused by deletions or insertions, translocations leading to in-frame or frame-shifted RNA fusions, alternative isoforms caused by aberrant RNA splicing leading to partial or complete intron retention, novel exon junctions, etc. In these cases, agretopicity values are typically not calculated and may be reported as not applicable. This should be taken into consideration if prioritizing variants of mixed type using these values. Interpretation of agretopicity is primarily relevant when the mutant amino acid(s) involve anchor residues of the MHC \(^{180}\). |
| HLA naming conventions | Neoantigen characterization workflows should consistently adopt the widely used standards and definitions for communicating |
| HLA typing (class I vs II typing) | histocompatibility typing information. Briefly HLA alleles are named using an HLA prefix followed by a hyphen, gene designation, asterix separator, and four fields of digits delimited by colons (e.g. HLA-A*02:101:01:02N). The four fields (typically of two or three digits each) represent the allele group, specific HLA protein, synonymous changes in the coding region, and non-coding differences respectively. Several popular HLA typing bioinformatics tools only report two field HLA types. The first two fields are generally sufficient for pMHC binding affinity predictions as these describe any polymorphisms that influence the protein sequence of MHC. However, three field typing might be desirable for patient specific assessment of expression because even silent variations in the DNA sequence of the HLA locus may influence read assignments to specific alleles. |
genome, exome or RNA-seq data (**Table 1**). Several groups have now conducted comparisons between the results of these tools and clinical assay results and reported high concordance, particularly for class I typing. Class II typing remains challenging, with fewer tools available and poorer consistency between the results of these tools and clinical assays. Use of clinical typing results remains advisable for class II. As in other areas of neoantigen analysis, use of a consensus approach involving multiple tools has become a common strategy for increasing confidence in HLA typing results 259.

| HLA typing (selection of data type and samples) | Several options are available for input data when performing HLA typing from NGS data including DNA (exome or WGS) or RNA-seq data. RNA-seq data often exhibits highly variable coverage across the HLA loci potentially leading to variable accuracy in typing for each. Coverage data from exome data may vary depending on the exome reagent’s design (probes selected against HLA regions) and capture efficiency. Care should be taken to evaluate for sufficient read coverage for each HLA locus when assessing HLA typing confidence. WGS data may exhibit comprehensive breadth of coverage, but generally at the expense of overall depth of coverage (again coverage achieved for the HLA loci specifically should be evaluated). In addition to data type, there is also the choice of whether to perform |
HLA typing using data from the tumor itself or reference normal sample. The normal sample has the advantage that it should represent the germline HLA alleles present in both the initiating cells of the tumor as well as antigen presenting cells of the immune system (relevant for cross presentation). In many clinical and research workflows, the quality of genomic DNA may be higher in the normal sample than the tumor (often of FFPE origin). The genomic DNA of the tumor may also be complicated by aneuploidy that affects the HLA loci (important to observe but also potentially interfering with HLA typing). HLA typing using the tumor DNA data has the advantage that it may more accurately reflect MHC binding and presentation of neoantigens on the surface of the targeted tumor cells. However, it is important to note that HLA typing tools are for the most part not designed for de novo HLA typing, rather they seek to determine which of a list of known alleles best explain the sequence reads of a given data set. HLA typing tools also generally do a poor job of reporting HLA typing confidence. At present, identifying loss of expression or somatic mutation of an HLA allele in a tumor is perhaps best treated as a separate exercise from HLA typing. One strategy for choice of data for HLA typing is to use all datasets available (DNA and RNA, normal and tumor), note any discrepancies, and investigate them.

HLA expression &

Loss of expression of MHC molecules by HLA deletion (or
mutation (downregulation) and somatic mutation of HLA loci have both been identified as possible resistance mechanisms for immunotherapies. It is therefore desirable for neoantigen characterization workflows to incorporate examination of HLA expression and somatic mutation in the tumor. Unfortunately, very few tools and best practices exist for this area. Given the sequence diversity of the HLA loci across individuals, when estimating expression of HLA transcripts in a tumor, it is desirable to customize the reference transcripts used (e.g. from the IPD-IMGT/HLA resource) to each individual’s HLA type by using the results of HLA genotyping to select the matching transcript sequences (three field matched) for expression abundance estimation (e.g. with Kallisto).

<table>
<thead>
<tr>
<th>Class I versus class II allele specification to binding prediction algorithms</th>
<th>Class I HLA alleles are typically supplied to binding affinity prediction algorithms using a standard two field format (e.g. HLA-A<em>02:01). However, class II alleles are often supplied as a pair using valid two field pairing combinations (e.g. DQA1</em>01:01-DQB1*06:02) to reflect the functional dimers of class II MHC. Peptide MHC prediction tools will typically document the syntax and list of valid pairings for which binding affinity predictions are supported.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proximal variation</td>
<td>Neoantigen selection pipelines often focus entirely on a single variant/position at a time and consider it to be independent of all nearby variations. It is important to carefully examine candidates to determine</td>
</tr>
</tbody>
</table>
if nearby variation exists that is both in phase (on the same allele) and close enough to influence the peptide sequence and therefore MHC binding predictions.  

| Peptide length considerations | Many human class I peptide MHC binding affinity prediction tools support a range of peptide lengths for each individual HLA allele (e.g. IEDB supports lengths 8-14 for class I for HLA-A*01:01). Typically while multiple lengths are supported, the peptides found to have strong binding will be highly biased towards the lengths actually favored by the allele (e.g. many human HLA alleles strongly favor nonamers). The open binding groove of MHC class II is thought to support a larger range of peptide lengths. This is reflected in some class II binding prediction tools though it should be noted that currently the IEDB API and web resource currently enforce a length of 15 amino acids only. |
| Relationship between genomic variants and short peptides | There is a complex relationship between genomic variants and the short peptide neoantigen candidates they might represent. Though rare, it is possible for multiple distinct somatic variations to result in the same amino acid change (e.g. several single nucleotide substitutions affecting a single triplet codon) and therefore might lead to identical neoantigens. If these were to occur on opposite alleles it might be important to analyze them separately because they could differ in expression level and/or their corresponding proximal variants, giving rise to distinct peptides. Other ways in which a single genomic variant can give rise to |
distinct short peptides for pMHC binding prediction, include the following. (A) A homozygous somatic variant representing two distinct alleles. If these alleles are in phase with one or more nearby heterozygous proximal variants distinct peptide sequences may result. (B) SNVs expressed in different RNA transcripts/isoforms that differ in their reading frame at the position of the variant, inclusion/exclusion of nearby alternative exons, or nearby use of alternative RNA splicing donor/acceptor sites. (C) Multiple short peptides will result simply from shifting the “register” of the somatic variant in a short sequence or from use of multiple peptide lengths (e.g. 8-11-mers) during pMHC binding affinity prediction. In some ways, mostly similar peptide sequences do not matter in peptide vaccine design since ultimately a longer peptide will incorporate several of them into a single peptide sequence. However, pMHC binding prediction algorithms require that you supply a short sequence, of a specific length, with the variant in a particular register and each of these and lead to different predicted binding affinity values. Making decisions about how to summarize, collapse, filter, and select representatives is one of the complexities that pipelines such as pVACtools address.

| Importance of transcript annotation quality | Peptides for neoantigen consideration are generally derived from the anticipated open reading frame of a known or predicted transcript sequence. A common consideration in variant effect annotation is |
and choice to select a single transcript variant annotation whether to allow annotations for each variant against multiple transcripts or to select a single representative transcript. If choosing a single transcript per gene, multiple strategies exist including the following. (A) Using a pre-selected automatically determined or manually curated choice of “canonical” transcript for each gene (B) Considering all transcripts but selecting the single transcript that results in the most confident and/or consequential predicted functional impact. This is the basic intent of the “--pick” option of the Ensembl Variant Effect Predictor (VEP) which chooses one block of annotations per variant using an ordered set of criteria (refer to the VEP documentation for extensive details). The benefit of choosing a single transcript for annotation of each variant is simplicity and in many cases will result in selection of a suitable peptide sequence for neoantigen analysis. However, the downside is that distinct peptides may not be considered and the peptide corresponding to the selected annotation is not guaranteed to be the best.

Note that a single variant may be assigned annotations for: multiple genes, multiple transcripts of the same gene, and multiple effects for the same transcripts. For example, a single variant can be annotated as splicing relevant near the edge of an exon causing exon skipping and also missense causing a single amino acid substitution. The same variant could be silent for a different transcript of the same gene and have a
regulatory impact on a transcript of another gene. Making sensible automated choices about how to choose and report neoantigen candidates that correspond to these variants is a complexity that neoantigen characterization workflows seek to address.

| Importance of transcript annotation quality | When using VEP it can be important to consider the Transcript Support Level assigned by Ensembl. As described above, this classification is one of many factors considered in choosing a single “best” transcript for annotation of variants. Occasionally a variant annotation will be reported with a dramatic effect (e.g. nonsense) but on further inspection this effect is only true for a transcript that is poorly supported by sequence evidence and another more reliable transcript would lead to different candidate neoantigen sequences. |
| Selection of pMHC binding affinity prediction cutoff(s) | Many peptide MHC binding prediction tools report binding strength as an IC50 value in nanomolar (nM) units. Peptides with a binding affinity less than 500 nM are commonly selected as putative strong binding peptides. However, the widespread use of this common binding strength metric may give some a false sense of consistency. Trusting a simple cutoff of 500 nM from a single algorithm should be avoided but combining scores from multiple algorithms should also be pursued very cautiously. The range, median, and even shape of distribution of IC50 scores across algorithms varies dramatically, even when applied to exactly the same peptides. Further complicating the selection process, |
| Interpretation of binding affinity from multiple binding prediction algorithms | Given the variability in IC50 predictions across binding prediction algorithms, some neoantigen workflows involve use of multiple binding prediction tools and attempt to calculate or infer a consensus. Best practices for determining such consensus are poorly articulated at this time and limited gold standard independent validation data sets exist to evaluate the accuracy of divergent predictions. Unsophisticated but pragmatic approaches currently involve reporting the best score observed, calculating the median score, determining average rank values, or manually visualizing the range of predictions across algorithms for promising candidates and making a qualitative assessment. |
| Neoantigen candidate reporting, visualization, and final prioritization | Prior to final review of candidates, automated filtering of variants and peptides that do not meet basic criteria (VAFs, binding affinity etc.) is performed to provide a more interpretable result. As discussed above, a single genomic variant can lead to many candidate peptide sequences (due to alternative reading frames, peptide lengths, registers, etc.). At the time of final candidate review/selection, a common strategy is to use... |
A pipeline that will automatically choose a single representative (best) peptide for each variant in a filtered result. Similarly, a condensed report may be generated to present only the most important information about each candidate. Final assessment of a candidate neoantigen may easily involve consideration of 20-50 specific data fields. Review of this data in spreadsheet form can be time consuming and inefficient and may make it difficult to consider some data in the context of a cohort of comparators (e.g. expression values are often best interpreted relative to reference samples). Tools are now emerging to facilitate more efficient visual interfaces for neoantigen candidate review (e.g. pVACviz).

| Vaccine manufacturing strategy | In the case of personalized cancer vaccine trials, the method of vaccine delivery can influence bioinformatics tool selection and other analysis considerations. For example, if candidates are to be encoded in a DNA vector, a tool such as pVACvector of pVACtools may be used to determine the optimal ordering of peptide candidates. Due to the combinatorial nature of candidate peptide sequence ordering, and the need to examine all pairs for junctional epitopes, this is currently one of the most computationally expensive and time consuming steps of these workflows. Similarly, if peptides are to be synthesized for a peptide vaccine this creates a need to predict possible problems with synthesizing each peptide (e.g. calculating “manufacturability” scores). |
Table 1.3 MHC class I binding algorithm comparison

A direct comparison of a subset of popular MHC class I binding predictors to show their variability in algorithmic structure, training data, supported HLA alleles and valid peptide lengths.

<table>
<thead>
<tr>
<th>Features/Softwares</th>
<th>Algorithm type used</th>
<th>Type of Data Trained on</th>
<th># of HLA alleles trained on</th>
<th>HLA alleles &amp; peptide length able to predict</th>
<th>Output information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pickpocket (2009)</td>
<td>Position specific weight matrices</td>
<td>In vitro quantitative binding data (&gt;150,000 data points)</td>
<td>More than 150 different MHC molecules</td>
<td>HLA-A, B, C, E, G alleles and for non-human primates, mouse, cattle and pig as well. Peptides of 8-12 in length.</td>
<td>Prediction values are given in nM IC50 values</td>
</tr>
<tr>
<td>NetMHCcons (2012)</td>
<td>Integration of NetMHC 3.4, NetMHCpa</td>
<td>In vitro binding affinity data</td>
<td>NetMHC 3.4 (94 MHC class I alleles),</td>
<td>Can predict peptides to any MHC molecule of</td>
<td>Prediction values are given in nM IC50 values and as % Rank to a</td>
</tr>
<tr>
<td>System</td>
<td>Method</td>
<td>Input</td>
<td>Output</td>
<td>Score</td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>n 2.8 and PickPocket 1.1</td>
<td>NetMHCpan 2.8 (&gt; 120 different MHC molecules), PickPocket 1.1 (94 different MHC alleles)</td>
<td>known sequence. Peptides of 8-15 amino acids in length</td>
<td>set of 200,000 random natural peptides</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NetMHC 4.0 (2016)</td>
<td>Artificial Neural Networks</td>
<td>In vitro binding affinity data</td>
<td>81 different Human MHC alleles (HLA-A, -B, -C, -E) and 41 animal alleles</td>
<td>Core position for binding within peptide, interaction core sequence, affinity in nM, rank of prediction compare with 400,000 random natural peptides (strong binders %rank</td>
<td></td>
</tr>
<tr>
<td><strong>NetMHCpan 4.0 (2017)</strong></td>
<td>Artificial Neural Networks</td>
<td>Binding affinity (&gt; 180,000 data points) and Eluted Ligand (MS) data</td>
<td>172 human and other animal MHC molecules</td>
<td>Can predict peptides to any MHC molecule of known sequence</td>
<td>Core position for binding within peptide, interaction core sequence, affinity in nM, rank of the predicted affinity compared to a set of random natural peptides (strong binders %rank &lt;0.5) etc.</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------</td>
<td>---------------------------------</td>
<td>---------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td><strong>MHCnuggets (2017)</strong></td>
<td>Gated recurrent neural networks</td>
<td>IC50 values from immunofluorescent binding experiments for pMHC Class I pairs (137,654 data points)</td>
<td>106 unique MHC alleles</td>
<td>Any MHC alleles, more reliable for alleles that are present in IEDB, Any peptide length is valid</td>
<td>IC50 binding affinity prediction</td>
</tr>
</tbody>
</table>
| **MHCfluury**  
**2018** | Allele-specific feed forward neural networks | Binding affinity and Eluted Ligand (MS) data (> 230,735 data points) | Across 130 alleles from IEDB combined with benchmark dataset from Kim et al \(^{260}\) | 112 alleles showed sufficient performance to include in predictor. Peptide lengths 8-15 are supported. | Affinity given in nM, percentile predictions across the models and quantile of affinity prediction among large number of random peptides tested |
| **EDGE**  
**2019** | Deep Neural Network | Peptide sequences from HLA immunoprecipitation followed by Mass Spectrometry characterization | Not explicitly specified | 53 HLA alleles, 8–15-mer (inclusive) | Not explicitly specified |
Appendix 2: Chapter 2 Supplemental Materials

Additional supplemental materials for Chapter 2 can be found at: http://www.science.org/doi/10.1126/sciimmunol.abg2200

A list of supplemental files are as following:

Data file S1. Seed dataset of strong binding neoantigen candidates
We identified 609,807 strong binding peptides for 328 HLA alleles from clinical and TCGA datasets. These served as a seed dataset for our anchor prediction workflow where a maximum of 10 peptides were selected at random for each HLA-peptide length combination. Peptide sequences, corresponding strong binding HLA alleles and the predicted median binding affinities across up to 8 different algorithms are included. The full dataset with all individual algorithmic prediction scores is also made available at http://genomedata.org/anchor_predictions/.

Data file S2. Anchor predictions for 328 HLA alleles
Prediction results from our computational workflow sorted by HLA alleles and peptide lengths. Overall scores are listed by peptide positions and represent the level of binding affinity change when mutated at the particular location. The 9-mer peptide data section contains additional information matching individual HLA alleles to their color-coded cluster in Figure 3.

Data file S3. HLA Summary Information
Information regarding all HLA alleles analyzed, including: 1) Nearest neighbor as defined by NetMHCpan, 2) Distance to nearest neighbor, 3) mean standard deviation of scores across all algorithms predicting for that allele using all available peptides in the seed dataset, 4) anchor cluster predicted as shown in Fig. 3., 5) training data available for all HLA alleles based on NetMHCpan4.0 (both elution and binding data) 6) number of algorithms able to generate predictions for each HLA allele, and 7) the detailed list of algorithms included in the previous column count. Note that if an HLA allele had training data, then it did not need to use a nearest neighbor for estimating binding and hence distance is 0 with nearest neighbor listed as itself.

Data file S4. HLA PDB data table
All protein data bank structures, collected for orthogonal validation of anchor prediction results, are listed. Table includes: the specific HLA allele and peptide pair, the PDB identifiers, predicted binding affinities and anchor cluster codes. The first tab of the spreadsheet contains information for structures collected that only contain the HLA-peptide complex and the second tab contains those collected that contain HLA-peptide-TCR complexes.
Data file S5. Orthogonal validation correlation data
A subset of X-ray crystallography structures were used in demonstrating the distribution of correlation scores for the distance and SASA metrics against prediction scores. Information on the subset data used such as peptide sequence, HLA allele, distance/SASA correlations and respective p values are included. The first tab of the spreadsheet contains information on the subset plotted from HLA-peptide only structures and the second tab contains information on the subset plotted from HLA-peptide-TCR complexes.

Data file S6. Summary of all in vitro and cell based experimental validation data
Validation experiments were performed on a total of 136 peptide-HLA combinations. These experiments include both IC50 binding assays (measured binding affinity) as well as cell-based stabilization assays (average MFI value). A summary of the entire dataset as well as information on predicted binding affinities and measured binding categories are included.

Data file S7. Breakdown of individual algorithm predictions and their correlation with validation data
Individual algorithm predictions (across 8 different algorithms) and their correlation with data from IC50 binding assays (measured binding affinity) were calculated. The spreadsheet includes two sheets, one with the individual algorithm scores for each peptide-HLA combination and the other with Pearson correlation coefficients for each algorithm across HLA alleles.

Data file S8. List of TCGA samples for impact analysis
A subset of TCGA samples were chosen using HLA-balance-based selection for our overall cohort-level impact analysis (Tab 1). An additional 100 TCGA samples were selected at random to further evaluate patient-level impact of anchor considerations (Tab 2). Specific TCGA sample names are included.

Movie S1. Demonstration of orthogonal validation using distance and SASA metrics from x-ray crystallography structures
Video of the X-ray crystallography structure of HLA-B*08:01 and 9-mer peptide FLRGRAYGL. The video highlights the three components of the complex, including HLA (green), peptide (pink) and B2M (blue). It also provides a zoomed in view of the MHC binding groove, showcasing atoms surrounding the peptide in spheres and sticks for spatial perspective.
References


166. Rubinsteyn, A., Hodes, I., Kodysh, J., and Hammerbacher, J. Vaxrank: A computational tool for designing personalized cancer vaccines. 10.1101/142919.


for the Analysis of Molecular Dynamics Trajectories. Biophys. J. 109, 1528–1532.


225. Murphy, C., and Elemento, O. AGFusion: annotate and visualize gene fusions. 10.1101/080903.


233. Bhattacharya, R., Tokheim, C., Sivakumar, A., Guthrie, V.B., Anagnostou, V., Velculescu, V.E., and


BMC Bioinformatics \textit{15}, 241.