nature communications

Article

https://doi.org/10.1038/s41467-024-54887-7

Towards designing improved cancer immunotherapy targets with a peptide-MHC-I presentation model, HLApollo

Received: 25 September 2023

Accepted: 25 November 2024

Published online: 30 December 2024

Check for updates

William John Thrift[1,9], Nicolas W. Lounsbury[2,9], Quade Broadwell[1,9], Amy Heidersbach[3], Emily Freund[3], Yassan Abdolazimi[3], Qui T. Phung[4], Jieming Chen[2], Aude-Hélène Capietto[5], Ann-Jay Tong[5], Christopher M. Rose[4], Craig Blanchette[6], Jennie R. Lill[4], Benjamin Haley[3], Lélia Delamarre[5], Richard Bourgon[2,7], Kai Liu[1,8,9] & Suchit Jhunjhunwala[2,9]

Based on the success of cancer immunotherapy, personalized cancer vaccines have emerged as a leading oncology treatment. Antigen presentation on MHC class I (MHC-I) is crucial for the adaptive immune response to cancer cells, necessitating highly predictive computational methods to model this phenomenon. Here, we introduce HLApollo, a transformer-based model for peptide-MHC-I (pMHC-I) presentation prediction, leveraging the language of peptides, MHC, and source proteins. HLApollo provides end-to-end treatment of MHC-I sequences and deconvolution of multi-allelic data, using a negative-set switching strategy to mitigate misassigned negatives in unlabelled ligandome data. HLApollo shows a 12.65% increase in average precision (AP) on ligandome data and a 4.1% AP increase on immunogenicity test data compared to next-best models. Incorporating protein features from protein language models yields further gains and reduces the need for gene expression measurements. Guided by clinical use, we demonstrate pan-allelic generalization which effectively captures rare alleles in underrepresented ancestries.

Antigen presentation by MHC class I molecules (MHC-I) is crucial to alert the adaptive immune system to the presence of non-self moieties. Recognition of somatically mutated peptides (neoantigens) by CD8 + T cells on the tumor cell surface drives anti-tumor immunity[1–5]. Recently, individualized neoantigen specific therapies (iNeST) have been developed to target neoantigens presented at the tumor surface[6–9]. This approach relies on accurate prediction of peptide presentation by MHC-I for identification of patient-specific personalized neoantigens.

The antigen presentation pathway and machinery for MHC-I consists of several components, including peptide processing by the proteasome, and other cytosolic and endoplasmic reticulum (ER)-resident proteases, transport of peptides to ER through specialized complexes, and peptide loading onto MHC-I aided by chaperones[10–13]. MHC-I presented peptides (ligands), typically 8-14 amino acids long, can be very diverse in their sequence because of the polymorphic nature of the MHC locus, with thousands of possible MHC-I alleles being present in the population. Besides, the locus is also polygenic, with 3 genes encoding the alpha chain in humans (HLA-A, HLA-B and HLA-C). Immunopeptidomics studies have shown that each of an individual's 3-6 MHC-I allotypes can bind and present

[1]Early Clinical Development Artificial Intelligence, Genentech, South San Francisco, CA, USA. [2]Oncology Bioinformatics, Genentech, South San Francisco, CA, USA. [3]Molecular Biology Department, Genentech, South San Francisco, CA, USA. [4]Microchemistry, Proteomics and Lipidomics, Genentech, South San Francisco, CA, USA. [5]Cancer Immunology, Genentech, South San Francisco, CA, USA. [6]Protein Chemistry, Genentech, South San Francisco, CA, USA. [7]Computational Science, Freenome, South San Francisco, CA, USA. [8]Artificial Intelligence, SES AI, Woburn, MA, USA. [9]These authors contributed equally: William John Thrift, Nicolas W. Lounsbury, Quade Broadwell, Kai Liu, Suchit Jhunjhunwala. Richard Bourgon, Kai Liu: Work Performed while at Genentech. e-mail: vincentliuk@gmail.com; suchitj@gene.com

up to 10,000 unique peptides forming a distinct ligand repertoire[14–17].

Given the high diversity of MHC allotypes and the entire ligand repertoire (ligandome), it has been a long-standing interest in the scientific community to understand and computationally model the ligandome. One of the earliest approaches to do so employed position specific weight matrices to identify peptide binding motifs[18]. In the third decade since then, several advanced and performant approaches have been developed, including the neural network approach implemented in NetMHC's pioneering suite of tools[19–21]. An explosion of ligandome data obtained using liquid chromatography–tandem mass spectrometry has allowed development of several advanced deep learning based methods (Supplementary Table 1). However, several heuristic choices continue to be made in these methods to deal with data complexity. For example, some methods are a mix of allele-specific models instead of a pan-allele generalized model that can predict for untrained alleles. Multi-allelic (MA) data, wherein there is no experimentally assigned association between a specific allotype and ligand, need to be deconvolved. Some methods only use single-allele (SA) data from engineered mono-allelic cell lines, while other methods conduct an upfront heuristic deconvolution of MA data that is not modeled. Given the large amount of training data and the capability of deep learning approaches, especially of attention-based approaches[22–24], we aimed to develop a more generic and performant approach, avoiding heuristics.

In this work we present HLApollo, a pan-allelic, transformer-based model for predicting peptide presentation by MHC-I. The model performs end-to-end modeling of MA deconvolution (by simultaneously processing all alleles for each peptide in a sample), peptide processing, and pan-allelic training. We demonstrate the importance of negative set switching, a training strategy and regularizer to mitigate the impact of falsely presumed negatives, on achieving high performance. Our baseline HLApollo, which uses only amino acid sequences of the peptide, flanks and MHC-I, achieves the best performance on presentation (12.65% average precision (AP) over BigMHC[25]), CD8 T cell response (4.1% AP over NetMHCPan4.1-EL, Wells et al. dataset[26]), and study holdout presentation datasets (7.9% more peptides recovered than SHERPA-EL[27]). When gene expression information is available, an extended HLApollo + expression model achieves an AP of 23.93% greater on a negative-peptide-augmented test set than the next best comparable approach, HLAthena[28]. We develop a strategy of adding gene-level presentation-relevant features from protein language models, which improves performance on CD8 T cell response (0.5% AP over baseline HLApollo, Schmidt et al dataset[29]). Finally, we introduce a linear regression model for predicting out-of-training allele performance using only a priori information and no training data. This predicts an increase of HLApollo genotype coverage on several underrepresented ancestries, raising the number of covered alleles by 791, and can guide future ligandome collection to be more equitable.

Results

Development of a comprehensive immunopeptidomics database

Though several public ligandome databases of peptide-MHC-I complexes (pMHC-I) exist, we needed to develop a schema and database that adequately reflects the workflow of a typical immune peptidomics experiment[30–33]. Thus, we developed mhcDB, a SQLite database of ligandome and paired bulk RNAseq data that enables quality control at the granularity of sample runs, harmonization of different data modalities, and meta-analyses of the original experiments. The schema provides strict tracking of metadata (e.g. study, donor/sample, HLA genotypes, analyte, treatment, fragmentation technique, search parameters, etc.) and biophysical data (e.g. peptide sequence, standardized peptide modifications, protein mapping, etc.) (Fig. 1a; Supplementary Fig. 1).

We assembled a total of 953,693 unique {peptide, genotype} tuples across 347 unique HLA-I genotypes (SA + MA at 4-digit resolution), 305,646 unique peptides and 171 unique HLA-I alleles from a set of 22 published studies and IEDB[34] (Fig. 1c, Supplementary Data 1, Supplementary table 3). Data were filtered to retain WT peptides containing only canonical amino acids (no post-translational modifications) and a non-redundant set of studies in our database (Supplementary Fig. 2). Our ligandome data covers 8 cancer indications, while B cell lines, healthy tissue, brain cancer, colorectal cancer and skin cancer together account for ~77% of the data (Fig. 1b). We also collected available bulk RNAseq data from matched samples, participants or healthy tissue, amounting to 43.8% of our ligandome data (Fig. 1c). We provide Supplementary Table 2 to track the datasets used to train and evaluate various models; the flagship benchmark (BM) dataset described above is referred to as BM1.

BM1's (SA only) average positional probability matrix taken across all alleles reveals patterns in line with previous studies[14,27,28]. In particular we note enrichment of hydrophobic/polar residues at the terminal anchor and acidic residues at auxiliary anchor positions 2–4 (Fig. 1d). Interestingly, the A/R/K 'cleavage signal' is enriched at both C1 and P1 (Fig. 1d), and longer peptides tend to favor glycines between the peptides and are enriched towards protein terminals (Supplementary Fig. 7).

In order to characterize the surface ligandome of our samples we plotted the frequency of the top-presenting genes along with properties such as expression, average protein length, and the distribution of presented peptides (Supplementary Fig. 3). Unsurprisingly, most of these genes are consistently highly expressed and/or encode long proteins, contributing to their high degree of presentation. Notably, 3 of the top 4 genes (*VIM*, *COL6A3*, *DYNC1H1*) consistently present across several indications and are necessary for the structural integrity of the cell. Interestingly, *VIM*, which is crucial for epithelial-to-mesenchymal transition[35,36], was the top presenting gene in multiple samples.

HLApollo outperforms state of the art pMHC-I prediction methods

Our goal was to leverage various techniques in pMHC-I modeling, including inductive bias negative generation, peptide processing, deconvolution, and pan-allelic generalization, to create a comprehensive model that outperforms less integrated approaches. Figure 2a depicts a schematic representation of our pan-allelic pMHC-I transformer[22] architecture, HLApollo. HLApollo separately processes the peptide and MHC-I pseudosequences[20] before both representations are concatenated for further processing, ensuring good individual representations before relevant pMHC-I interactions are processed. To reflect the rarity of peptide presentation, we used a 1:99 ratio of positives:negatives in the test set, and used the AP metric for performance evaluation for this skewed ratio. HLApollo achieved an AP of 74.83% (74.67%-74.99% within one standard deviation) through 5-fold cross validation (see "methods") on our test set BM1 that is based on a 10% hold out of SA/MA elution data without any pMHC-I overlap with the training data (Supplementary Data 1). For model performance over peptide lengths, we have provided Supplementary Fig. 4.

An important challenge of pMHC-I models is (over)fitting to falsely presumed negative peptides, which can occur in the training data due to random sampling of the reference proteome for generating training set negatives, as also noted by other methods like NN-align[37], BigMHC[25], etc. This is akin to the problem of positive-unlabeled, i.e. PU learning[38]. In essence, the randomly sampled peptides are unlabeled, and we declare them as negatives on the basis that only a small fraction of them might actually be positives. Nevertheless, some overfitting to the negative data might still occur. To overcome this issue, we used negative set switching, a training strategy where we sample a new exclusive set of negatives (out of 500 candidate sets) after each epoch of training. The approach is similar to that of NNAlign_MA[39], where

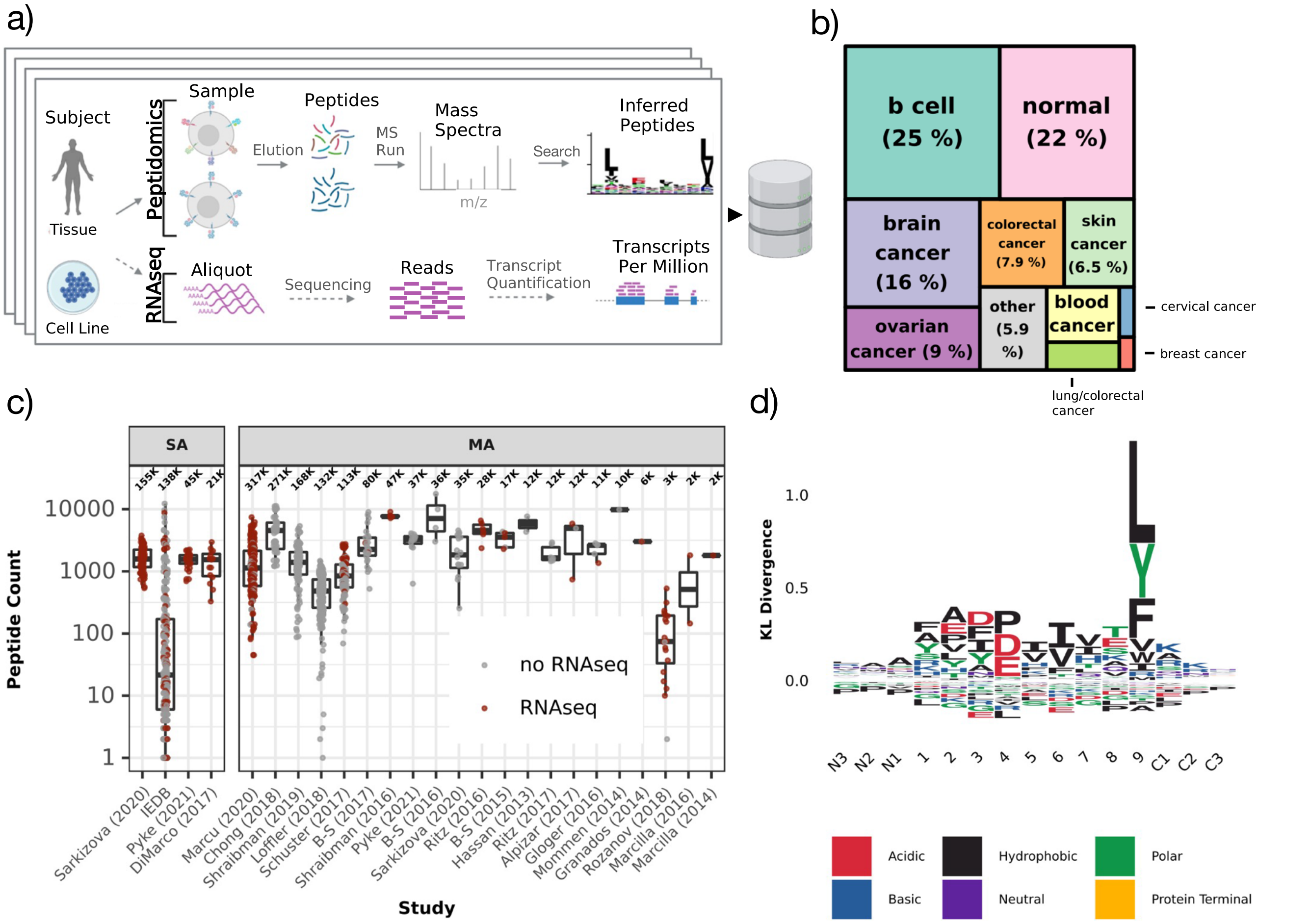


Fig. 1 | Summary of immune peptidomics data. a Schematic of immunopeptidomics workflow and/or matched bulk RNAseq workflows for a given study. Data from 22 studies and IEDB was processed and imported into a sqlite database, mhcDB. Figure 1a was created with BioRender.com, released under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International license. **b** Treemap plot showing the proportion of tissue types or disease indications across mhcDB. The number of (epitope,genotype,sample) tuples are as follows (b cell: 431,346; normal: 374,914; brain cancer: 265,550; colorectal cancer: 135,339; skin cancer: 111,796; ovarian cancer: 154,483; other: 101,521; blood cancer: 73,080; lung/colorectal cancer: 37,196; cervical cancer: 13,513; breast cancer: 8,902). **c** Unique peptide counts per sample from studies of single-allele (left) and multi-allele (right) data type, respectively. Samples with matched RNAseq are colored red. All extensions were based on simple logistic regression using the baseline HLApollo score and the additional feature. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend up to 1.5x the interquartile range (IQR = Q3 − Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly. $n = 953{,}693$ unique {peptide, genotype} tuples across $n = 347$ unique HLA-I genotypes (single allelic (SA) + multi allelic (MA) at 4-digit resolution), $n = 305{,}646$ unique peptides and $n = 171$ unique HLA-I alleles from a set of $n = 22$ published studies and IEDB d) Position-wise amino acid enrichment at the N-terminal flank, peptide (positions 1-9), C-terminal flank for 9-mer peptides, normalized by the number of peptides per HLA genotype (71,109 positive peptides; 136,680 negative peptides; 135 alleles) Kullback-Leibler (KL). Source data are provided as BM1.tar.gz, fig1c.csv.

both positive and negative data are sampled from the full training data prior to each training epoch. This prevents the model from observing false negatives more than once, and is an essential regularization strategy for our model. The impact of the number of negative sets on performance is depicted in Fig. 2b. Overfitting was observed with ten or fewer negative sets. In all, the five-hundred-set negative set switching used here led to an improvement of 11.79% AP compared to one negative set, among the largest contributors in performance in the ablations we investigated.

To further quantify the sources of performance gains, we present in Fig. 2c the performance of HLApollo (without ensembling) and various ablations on BM1. First, we consider pan-allelic training. Recently, HLAthena[28] reported that a pan-allelic approach actually performed worse than individual models for their feed forward neural network (FFNN) model, especially for 8-mers. In contrast, HLApollo was benefited by pan-allelic training across all peptide lengths, increasing AP by 13.9% (Figs. 2c, Supplementary Fig. 4). Thus, a pan-allelic approach was largely beneficial to pMHC-I presentation prediction in our approach. Below we also show pan-allelic generalization, which points to the reuse of representations across alleles.

Peptide processing information is expected to improve pMHC-I presentation prediction, as not all peptides might be processed from their parent protein. HLApollo jointly learns peptide processing, which we achieve by concatenating the peptide and flanking sequences (up to 5 residues) prior to being fed into the model. We further employ a regularization strategy: random removal of some or all of the flanking residues, to ensure good predictions on flanked, partially flanked, and flankless cases (see "methods"). End to end treatment of peptide processing leads to a 0.9% increase in AP.

Multiallelic (MA) data comprises more than twice as many peptides as SA data in our dataset, so it is important for models to learn effectively from such data. There are several ways to handle MA data, including upfront deconvolution and then using deconvolved SA data[27,40], or model-based deconvolution[39]. Some approaches only use

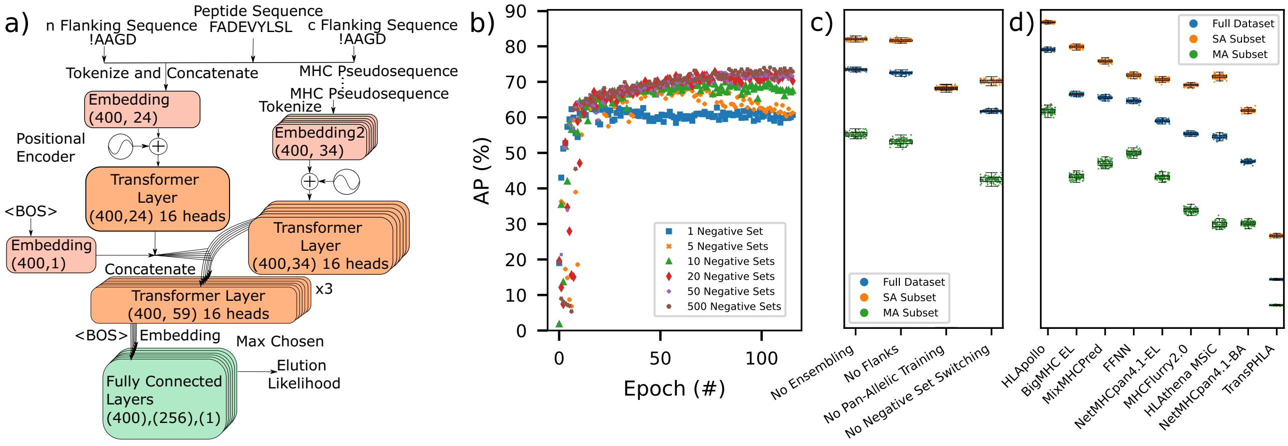


Fig. 2 | HLApollo model architecture and performance. HLApollo performance was assessed using dataset BM1 (19,740 test positives, see Supplementary table 2 for details on the dataset). **a** A schematic of the transformer-based model used throughout this work. Boxes correspond to neural network layers, the shape of the input to each layer is given in the box. The number of neurons in each of the 3 fully connected layers are given in the fully connected layers box. BOS:Beginning of sequence (**b**) Plot of the average precision over epochs for one, five, ten, twenty, fifty, and five hundred negative sets (blue square, orange x, green triangle, red diamond, purple cross, and brown circle, respectively) (**c**) AP values of non-ensembled HLApollo and various individual ablations (all ablations performed without ensembling or 5 fold cross validation). Blue dots depict the performance of one bootstrap (*n* = 100) on the full test set (multiallelic (MA) and single allelic (SA) data), while orange and green dots depict the performance subset to just the SA and MA data, respectively. **d** AP values of various contemporary models on the test dataset, and our implementation of a feed forward neural network (FFNN) model. Results bootstrapped, *n* = 100. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend up to 1.5x the interquartile range (IQR = Q3–Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly within a 0.5 width span. Statistical significance was assessed using the Wilcoxon signed rank test (two sided), of the performance bootstrapped 100 times with respect to HLApollo. All p-values with respect to HLApollo are all 3.9e-18, indicating perfect separation between all points from HLApollo, and shown as ***. Average Precision (AP). Source data are provided as fig2b.csv, fig2cd_full.csv, fig2cd_ma.csv.

SA data[25]. Like, NNAlign_MA[39], HLApollo continuously learns which MHC-I allele presents a peptide by sending all MHC-I pseudosequences through the model in parallel (Fig. 2a), and uses the allele with maximum likelihood to evaluate loss. This allows the model to improve deconvolution after every batch. This results in better performance on test MA data, as evidenced by a smaller gap between the SA and MA subsets of the BM1 test set for HLApollo than other models evaluated in Fig. 2d.

For comparisons with other contemporary models, depicted in Fig. 2d), we ensembled 10 models for our flagship HLApollo, which led to an AP to 78.7%, 12.65% larger than any other model considered on a test set with peptides held out from the training set in an allele-specific manner ($p = 3.9*10^{-18}$, Wilcoxon signed rank test, bootstrapped 100 times and paired by bootstrap). Of note, the gain in performance was maintained when peptides were also held out irrespective of allele specificity (Supplementary Fig. 8). In order to compare the use of a sequence-based model with FFNNs common in the literature[28,41,42], we trained our own FFNN, using a similar strategy as NetMHCpan. An FFNN architecture (with flank information, negative set switching, in-model deconvolution, and ensembling) as opposed to an attentive sequence based architecture led to a large decline in AP of 14.5% (Fig. 2d). Supplementary Fig. 9 depicts these results (and the results of Figs. 3d, and 5a–c) using an alternative metric, ROCAUC calculated up to 10% FPR.

Inclusion of gene expression and protein features reduces false discovery at low expression levels

Besides peptide biophysical properties, expression of the source gene has been reported to be an important factor in determining presentation[28]. Empirical peptide enrichment (EPE) showed a log-linear relationship with expression (Fig. 3a). Interestingly, we observed EPE and HLApollo logit scores crossing 0 simultaneously (Fig. 3a), suggesting that peptide enrichment is reflected appropriately by HLApollo. We also expect that expression especially reduces false positive calls in lowly or non-expressed genes. We switched to a 4999:1 negative:positive ratio test set to explore this effect (BM2). Due to the heterogeneity of expression values between samples we use sample-specific test sets to explore the impact of this effect. We then modeled the effect of expression by using a simple logistic regression approach, using HLApollo and expression values. This increased AP by 6.03% (Fig. 3d), demonstrating the benefit of including gene expression information when available.

In addition to expression, previous studies showed that gene-specific features, such as protein localization annotation, such as that from Gene Ontology (GO), and presentation bias scores provide marginal improvement to presentation prediction[14,27,28]. Here, we sought to capture a wide range of gene-specific features at once by using protein language models. We used Evolutionary Scale Model (ESM) 1b[43] to extract a gene-level embedding, from which a FFNN (depicted in Fig. 3b)) was trained to predict the presentation likelihood (ESM_{MHC-I}). Using ESM_{MHC-I} with HLApollo in a logistic regression improved AP, while adding GO to HLApollo did not. {HLApollo, expression} ourperforms {HLApollo, ESM_{MHC-I}} by 3.47%, with a p value of $7.97*10^{-19}$ based on a Wilcoxon signed rank test (paired by samples). ESM_{MHC-I} yielded a similar improvement to the FDR at low expression bins as expression itself (Fig. 3c), suggesting that the gene-specific features captured by ESM_{MHC-I} could be proxies for consistently lowly expressed genes (albeit with increased frequency of outliers at higher expression bins, see Supplementary Fig. 6). Regardless, these results show that ESM_{MHC-I} can be useful in applications where expression values are unknown, validated further by improved performance of HLApollo on an independent test data modality of T cell response, considered below.

Validation on tissue presentation holdouts and CD8-T cell response to neoantigens

Model generalization to unseen human tissue samples is paramount for clinical applications and challenged by the variety of biological/technical variation introduced therein. Figure 4a–c compares HLApollo's generalization to holdout human tissue studies (BM3) from

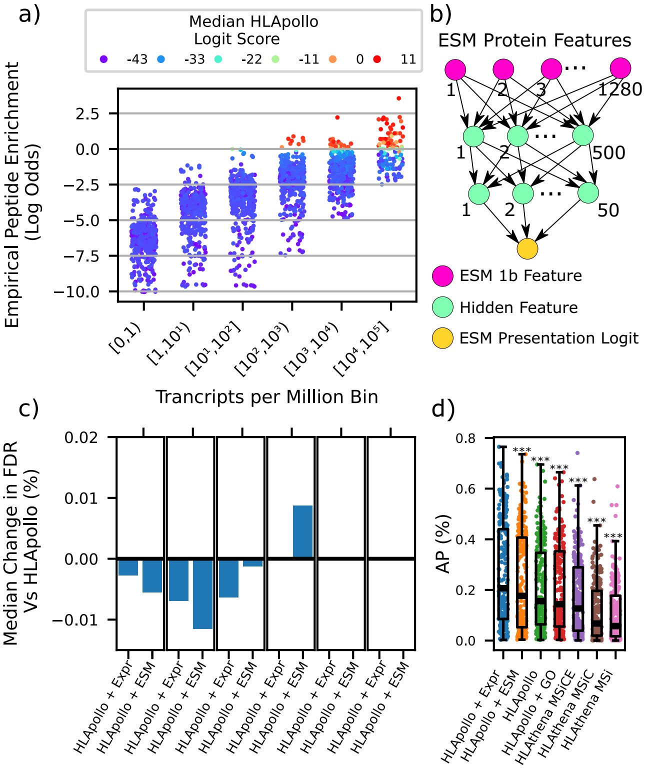


Fig. 3 | Gene expression or protein language model features boost HLApollo performance. The test dataset used is BM2 (BM1 subset to positives with expression, 4999:1 negatives:positives, 174,682 positives, see Supplementary Table 2) (**a**) Binned expression (TPM) vs. empirical peptide enrichment (see "Methods".) where each point represents a sample with a total of 50,000 peptides (combined positives and negatives). Samples (*n* = 217) are colored by the median HLApollo logit prediction in a given expression bin. **b** FFNN model used to extract presentation features from ESM1b protein features. **c** median per-sample false discovery rate (FDR) gain over HLApollo of the logistic regression models further adding expression (left bars) or ESM_{MHC-I} score(right bars), at increasing expression levels. 54550, 77139, 49827, 3149, 263, and 35 peptides are found in bins [0,1), $[1,10^2)$, $[10^2,10^3)$, $[10^3,10^4)$, and $[10^4,10^5]$, respectively. **d** depicts the median performance per sample (*n* = 217) of various extensions of HLApollo that utilize features beyond the sequence features considered in Fig. 2. All extensions were based on simple logistic regression using the baseline HLApollo score and the additional feature. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend up to 1.5x the interquartile range (IQR = Q3−Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly within a 0.5 width span. Statistical significance was assessed using the Wilcoxon signed rank test (two sided), of the performance bootstrapped 100 times with respect to HLApollo. All p-values were less than 10^{-3}, and shown as ***. *p* values with respect to HLApollo + Expr are: HLApollo + ESM 8.0e-19, HLApollo 1.2e-31, HLApollo + GO 3.1e-32, HLAthena MSiCE 7.e-25, HLAthena MSiC 4.0e-33, HLAthena MSi_rank 1.4e-34. Average Precision (AP), False Discovery Rate (FDR). Source data are provided as fig3a.csv, fig3c.csv, fig3d.csv.

Schuster et al.[44], Löffler et al.[45], and Pyke et al.[27], respectively, with recent pMHC-I presentation models. Here we compared the models' study-holdout performance by the fraction of peptides that are ranked better than 0.1% percentile rank (see "methods"), as in other works[27,28] (assessment was limited to this test set and sensitivity values available from Pyke et al. [27] because of limited availability of some of the comparator methods) HLApollo showed higher sensitivity than the other models considered in study holdout generalization, although due to small numbers of samples the difference is not consistently statistically significant compared to the second best model, with p values (Wilcoxon signed rank test paired by samples) of 0.055, 0.46, and $4.9*10^{-4}$ for Schuster, Löffler, and Pyke, respectively.

An important application of pMHC-I presentation models is to rank neoantigens for use in individualized neoantigen-specific cancer therapies. pMHC presentation is a prerequisite step in inducing a T-cell response to neoantigens, hence pMHC presentation prediction is also expected to enrich neoantigen candidates that induce a T-cell response. Notably, immunogenicity depends on other factors besides presentation, while we are using immunogenicity data as an orthogonal data modality (instead of ligandome data) for evaluating pMHC presentation prediction methods. To this end we considered cancer neoantigen immunogenicity datasets acquired / collected by Schmidt et al.[29] (BM4) and Wells et al. [26], (BM; Fig. 4a, b). HLApollo outperformed competing pMHC-I models on these immunogenicity data sets. We observe that HLApollo + ESM_{MHC-I} improves AP by 0.5% (p value $2.4*10^{-3}$, Wilcoxon signed rank test, bootstrapped 500 times and paired by bootstrap) and 2.0% (p value $1.9*10^{-77}$) for Schmidt et al. (Fig. 4a) and Wells et al. (Fig. 4b), respectively, compared to baseline HLApollo, while GO did not add any benefit. We also explored the effect of using flanking sequence information during training, on the performance of these trained models on immunogenicity data, since using flanks demonstrated improved performance for presentation prediction. However, using flanks during training did not show consistent performance gains on immunogenicity data sets-based validation (Supplementary Fig. 11).

To further compare our model with others on evaluating microbial antigen immunogenicity, we obtained a collated dataset of infectious disease antigens from Albert et al, originally sourced from all datasets available within IEDB (BM8, Supplementary Table 2; Fig. 5c). Here we only evaluate HLApollo, and not HLApollo + expression or ESM, as expression and protein features for infectious disease antigens are not applicable features. Interestingly, on this set we observe MixMHCpred outperforming HLApollo 77.8% to 77.1% AP (p value $4.6*10^{-77}$, calculated as with Fig. 5a), whereas it was among the lower-performing methods in cancer neoantigen immunogenicity datasets (HLApollo outperformed the third best model, MHCflurry-2.0 by 1% AP).

A bona fide deconvolution approach

Current deconvolution approaches for MA data use semi-automatic motif co-occurrence and exclusion principles[46], or substantially rely on SA data-trained models to untangle MA data[27,39,47]. In contrast, our strategy enables de novo training from MA data, by using multiple alleles simultaneously during training. We rigorously evaluated our deconvolution strategy using a synthetic MA test dataset (BM6), constructed from a SA dataset, so that the ground truth of the presenting allele in the MA data is known. Borrowing from co-occurrence and exclusion principles[46], we created 'deconvolvable' and 'non-deconvolvable' sets of genotypes (see "methods"). We compared a model trained on this synthetic MA dataset and one trained on the underlying SA data, evaluating both on a held-out SA test data on these alleles (BM6, Supplementary Table 2). Remarkably, the model trained on MA data alone performed nearly identical to the model trained on SA data, with no distinction observed between 'deconvolvable', and 'non-deconvolvable' alleles (Fig. 6a). Overall, the model calls 93.9% of alleles correctly, based on the allele with the highest score from the MA-only model (91.1% for 'non-deconvolvable' and 95.6% for 'deconvolvable').

While our deconvolution strategy works well in synthetic data, we also must see if systematic problems with certain alleles (e.g. low presentation of peptides by HLA-C) reduce deconvolution performance in practice for real data. One challenge of real data is that SA samples used for evaluation are also study holdouts. In order to disentangle performance decline caused by deconvolution and by generalization to new studies, we create a training dataset ('Sarkizova

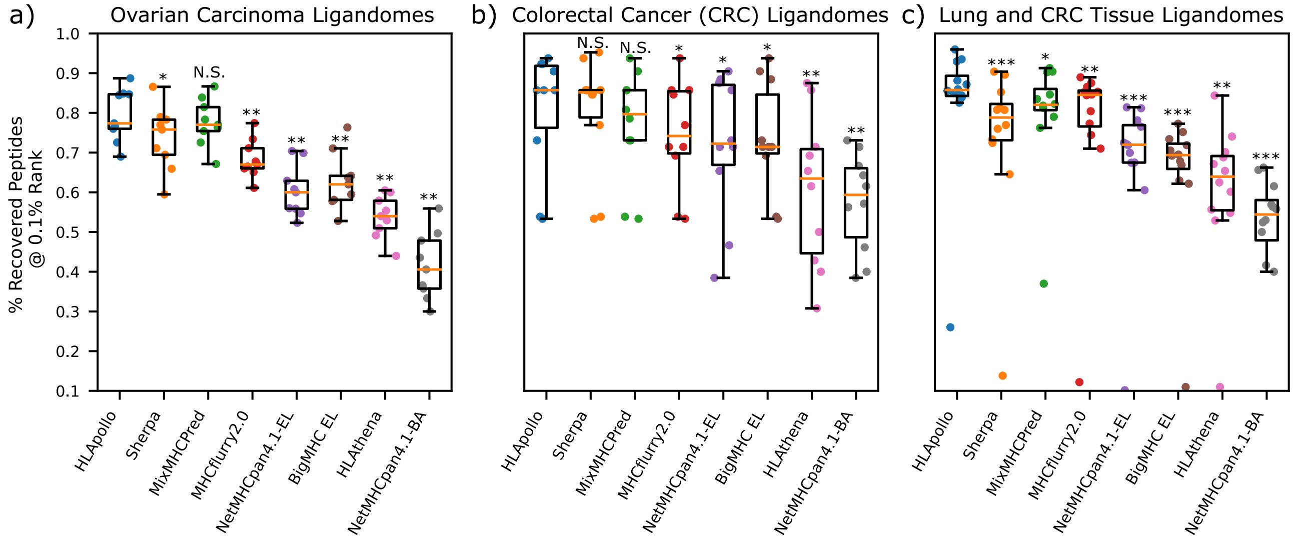


Fig. 4 | HLApollo shows superior performance in tissue holdout data. a depicts per sample boxplots of the fraction of eluted peptides ranked at less than 0.1% rank by various models for held out Schuster et al. ($n = 9$ samples), p-values vs. HLApollo are 2.2*10-01, 5.7*10-01, 4.7*10-03, 7.9*10-04, 1.5*10-03, 4.1*10-04, 4.1*10-04 for BigMHC EL, Sherpa, MixMHCPred, MHCflurry, NetMHCpanEL, HLAthena, NetMHCpanBA, respectively. **b** Loffler et al., ($n = 10$ samples) p-values vs. HLApollo are 6.7*10-01, 3.8*10-01, 1.9*10-01, 1.7*10-01, 2.5*10-01, 1.9*10-02, 8.0*10-03 for BigMHC EL, Sherpa, MixMHCPred, MHCflurry, NetMHCpanEL, HLAthena, NetMHCpanBA, respectively. **c** Pyke et al. ($n = 12$ samples). (BM3, Supplementary Table 2) p-values vs. HLApollo are 1.9*10-02, 6.9*10-02, 1.4*10-01, 4.8*10-04, 4.8*10-04, 8.1*10-04, 4.8*10-04 for BigMHC EL, Sherpa, MixMHCPred, MHCflurry, NetMHCpanEL, HLAthena, NetMHCpanBA, respectively. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend up to 1.5x the interquartile range (IQR = Q3 - Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly within a 0.5 width span. The asterisks depict the statistical significance of the performance distributions with respect to the best performing model, using the Wilcoxon signed rank test (two sided). Average Precision (AP). *:$10^{-2} < p \leq 10^{-1}$, **: $10^{-3} < p \leq 10^{-2}$, ***:$p \leq 10^{-3}$. Gene ontology (GO). Source data are provided as fig4-a.csv, fig4b.csv, fig4c.csv.

holdout dataset', Supplementary Table 2) based on BM1, but with all data from one study, Sarkizova et al.[28], held out and train a new model (Sarkizova Holdout model). We thus create two ablated models compared to the full HLApollo model: an MA-only model (SA data removed from training), and a Sarkizova Holdout model (one study data removed from training). Both would be expected to underperform compared to the full model. Figure 6c depicts the AP drop (relative to the full model) of the MA-only model and Sarkizova Holdout on the 71 SA alleles from Sarkizova et al. that exist in the MA data, but have SA alleles from sources other than Sarkizova (Fig. 6b, BM7). A median AP drop of 1.9% was seen for Sarkizova Holdout, likely due to additional information on these alleles that was lost after ablating the Sarkizova study from the training data. This is the baseline AP drop against which we can compare the larger expected AP drop of the MA-only model. The MA-only model indeed experienced a more significant median AP drop, of 15.3%. This drop is driven in large part by HLA-C (Fig. 6c), which declines by 21.1%, indicating that deconvolving HLA-C specificity from MA data is particularly challenging.

We further evaluated how deconvolution 'difficulty' impacted HLApollo's ability to recognize patterns from individual alleles observed in MA genotypes. We propose four levels of difficulty for deconvolution: trivial, easy, medium, and hard, based on their combinatorial uniqueness to MA samples. The definitions are depicted diagrammatically in Fig. 7a (see "methods"). Figure 7b depicts predicted (MA + SA trained) HLApollo motifs (upper) and observed motifs (lower) from easy, medium, and hard deconvolvable alleles. In order to obtain observed motifs for these alleles, we engineered SA cell lines and obtained ligandomes (see "methods" for more details), but did not include them in our training dataset. One may observe an improving trend of predicted motifs from easy to hard alleles, as expected. Yet, even for the hard allele, B*51:05, HLApollo predicts a reasonable motif and achieves an AP of 82%, indicating good deconvolution performance.

Pan-allelic generalization

In a clinical setting, full coverage of a participant's HLA-I genotype by pMHC-I presentation models is desired. Therefore it is critical to have a pan-allelic model with good generalization for untrained alleles. Yet no method currently exists to determine if a model will have good performance on a particular untrained allele. To validate HLApollo's pan-allelic generalization, we first observe the degradation of AP for alleles for a model trained without them in its training dataset (Fig. 8a; see "Methods"). Note that each allele required making a new fitted model this way. While AP degradation is observed, the median AP drop is 9.6%, 7.7% worse than the study holdouts compared in Fig. 6c, but still much better than other ablations and comparator models (Fig. 2). Supplementary Fig. 10. depicts the results shown in Fig. 8a with bubble size depicting the prevalence of the allele in its highest-prevalence ancestral group, showing good generalization even to rare alleles. Thus HLApollo shows pan-allele generalization, albeit with varying levels of performance degradation on unseen alleles.

To model performance expectation on unseen alleles, we use a modified approach based on HLAthena[28], producing a linear regression model to model AP from attributes of the dataset, but here predicting the out-of-training (OOT) performance. We find this simple model was predictive of untrained AP (Fig. 8b). The parameters used, their coefficients, and p values are shown in the upper half of Fig. 8f (see "Methods" for in depth description of the parameters). Unsurprisingly, 9-mer motif entropy was found to have the largest impact, followed closely by motif abundance and the BLOSUM distance of the OOT 9-mer motif's nearest neighbor (NN) in the trained dataset. The latter feature's importance indicates that HLApollo reuses representations that it has built for other alleles, a key feature of pan-allelic generalization.

While the model depicted in Fig. 8b is useful for understanding pan-allelic generalization, it has limited clinical applicability because many of the important features are not known a priori (as ligandome

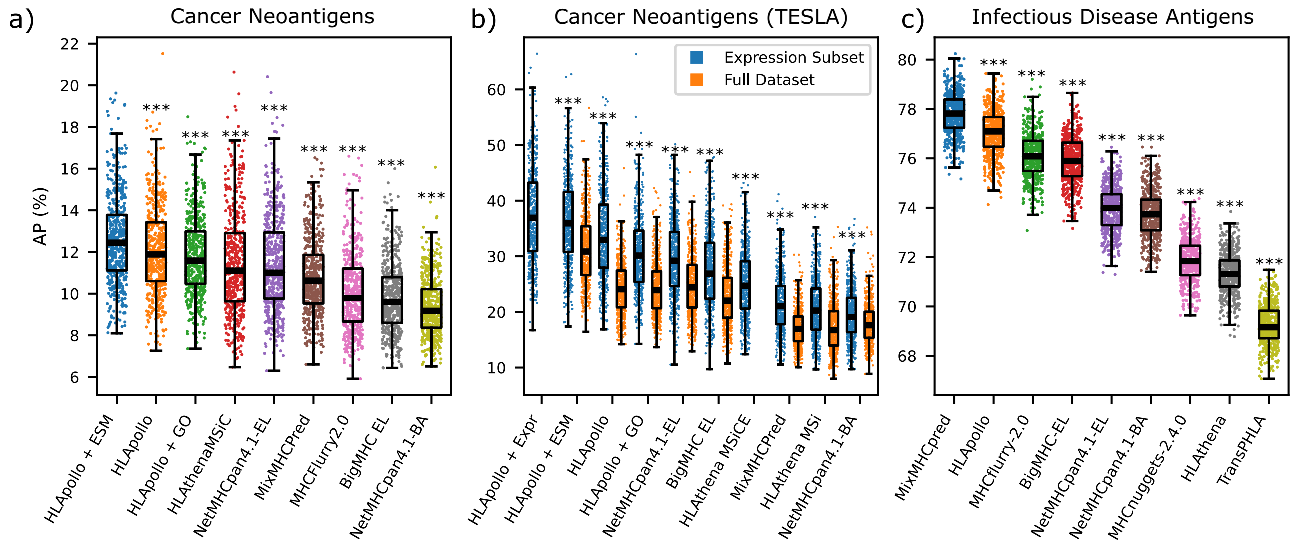


Fig. 5 | HLApollo shows superior performance in immunogenicity test data. a depicts the AP of various models on data sets of pre-existing CD8 T cell response to cancer neoantigens, acquired from Schmidt et al. (BM4, Supplementary Table 2) p-values vs HLApollo + ESM are 4.8*10-10, 1.7*10-27, 2.9*10-27, 2.1*10-29, 3.0*10-57, 8,9*10-27, 3.1*10-82, 3.5*10-81 for HLApollo, HLApollo + GO, HLAthena, NetMHCpanEL, MixMHCPred, MHCFlurry, BigMHC, NetMHCpanBA, respectively. 1. **b** AP of various models on Wells et al (BM5, Supplementary Table 2) after 500 bootstraps. p-values vs HLApollo + Expr are 3.3*10-5, 1.9*10-77, 4.0*10-82,1.6*10-61,1.5*10-81,2.6*10-70, 1.3*10-83, 8.7*10-82,4.1*10-83 for HLApollo + ESM, HLApollo, HLApollo + GO, NetMHCPanEL, BigMHC, HLAthena MSiCE, MixMHCPred, HLAthena MSi, NetMHCPanBA, respectively. **c** depicts the AP of various models on a dataset of CD8 T cell response to infectious disease antigens, acquired from Albert et al (BM8, Supplementary Table 2) after 500 bootstraps. *p*-values vs MixMHCPred are 4.6*10-77, 1.5*10-83, 1.3*10-83, 1.3*10-83, 1.3*10-83, 1.7*10-83, 1.3*10-83, 1.3*10-83 for HLApollo, MHCflurry, BigMHC, NetMHCPanEL, NetMHCPanBA, MHCNuggets, HLAthena, TransPHLA, respectively. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend up to 1.5x the interquartile range (IQR = Q3 - Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly within a 0.5 width span. The asterisks depict the statistical significance of the performance distributions with respect to the best performing model, using the Wilcoxon signed rank test (two sided). Average Precision (AP). *:$10^{-2} < p \leq 10^{-1}$, **:$10^{-3} < p \leq 10^{-2}$, ***:$p \leq 10^{-3}$. Gene ontology (GO). Source data are provided as fig5a.csv, fig5b_full.csv, fig5b_subset.csv, fig5c.csv.

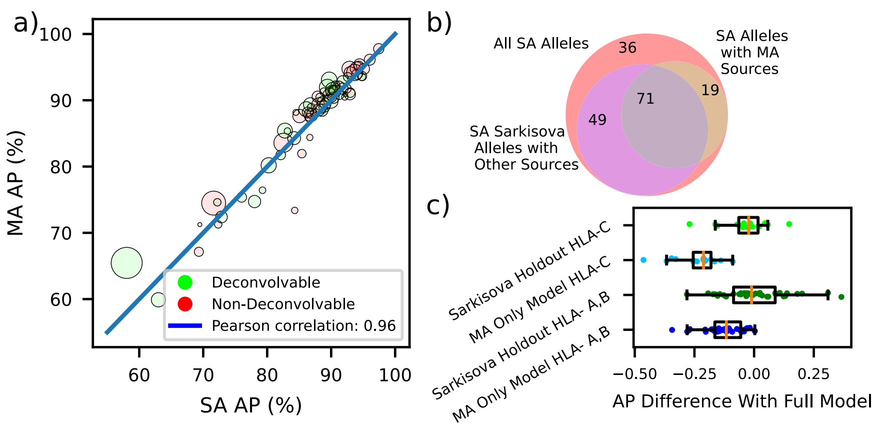


Fig. 6 | HLApollo performs de novo deconvolution on multiallelic (MA) data. a depicts a scatter plot of per allele AP on 72 test alleles obtained from a model trained using only synthetic MA data, and a model trained with just the single allelic (SA) version of the dataset. Both models are evaluated on the same SA dataset (BM 6, Supplementary Table 2). Alleles determined to be deconvolvable and non-deconvolvable according to the Bassani-Sternberg definition are plotted in green and red, respectively. **b** depicts a Venn diagram of the various sources of SA data available to train a model, all SA alleles (red), all SA alleles from Sarkizova et al. that also have other study sources (purple), and SA alleles with MA sources (brown-orange). The grey-blue circle represents SA alleles which exist in Sarkizova et al. and have other data sources, and which also have MA sources, this is used for evaluation in (**c**). **c** depicts boxplots of the AP drop relative to the full model evaluated on SA Sarkizova alleles with MA sources (BM 7, Supplementary Table 2). The models evaluated are: 1) Sarkizova holdout (a model trained without data from Sarkizova et al.) and 2) MA only model (a model trained with only MA data). Upper (light green, *n* = 17) depicts the performance drop of just HLA-C alleles of the Sarkizova holdout, upper middle (light blue, *n* = 17) depicts the performance drop of just HLA-C alleles of the MA only model, lower middle (green, n = 37) depicts the performance drop of the Sarkizova holdout on HLA-A,B, and lower (blue, *n* = 37) depicts the performance drop of the MA only model on HLA-A,B. The center line in the box plots represents the median; box limits indicate the upper (Q3) and lower (Q1) quartiles; the whiskers extend to 1.5x the interquartile range (IQR = Q3–Q1); and outliers are shown as the points outside of the whiskers. Additionally, a scatter plot overlay of every data point is included, with their x-coordinates jittered uniformly within a 0.5 width span. Average Precision (AP) Source data are provided as fig6-a.csv, fig6c.csv.

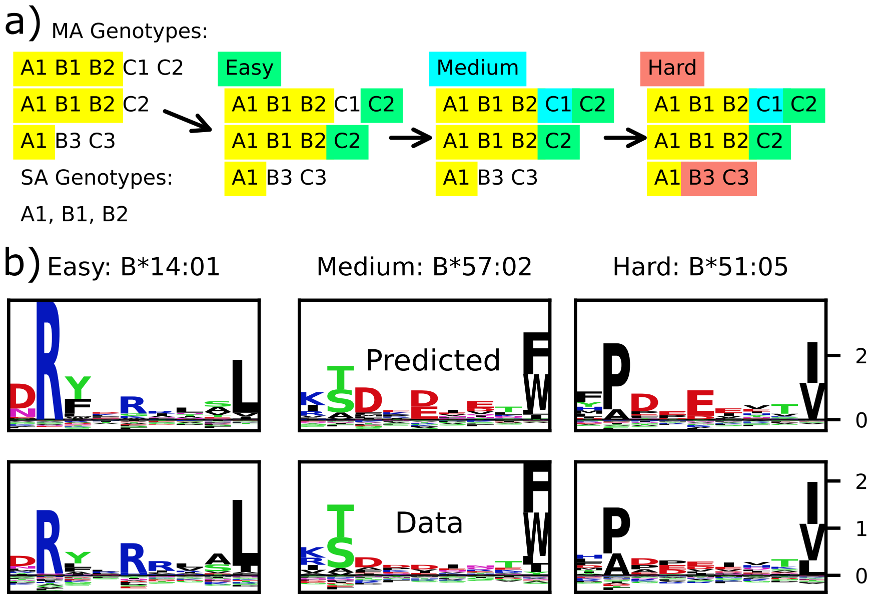


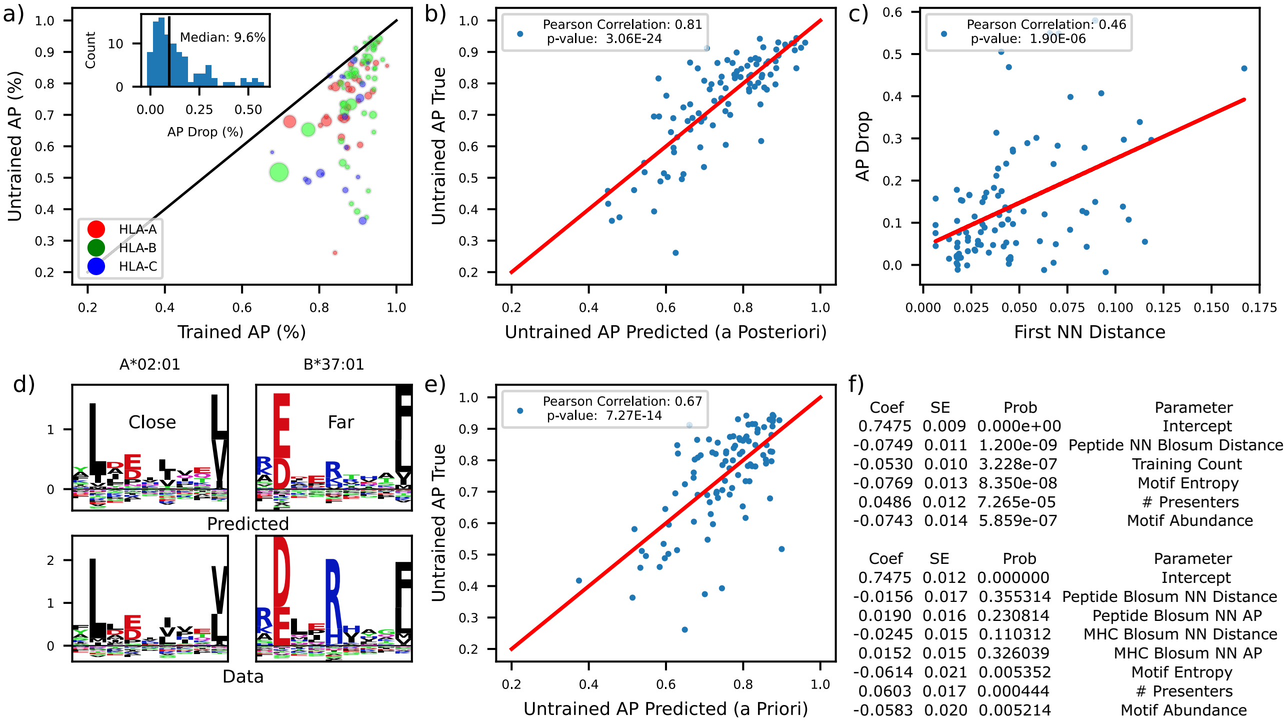
Fig. 7 | HLApollo performs deconvolution even in complex allele co-occurrence contexts. a depicts a diagram illustrating our method for allele deconvolvability difficulty, where yellow, green, blue, and red indicate trivial, easy, medium, and hard, respectively. **b** depicts motif plots for B*14:01, B*57:02, and B*51:05, which are easy, medium and hard to deconvolve, respectively, predicted by (SA and MA trained) HLApollo (above), and observed (below). HLApollo achieves and AP of 94%, 83%, and 82% for B*14:01, B*57:02, and B*51:05, respectively. Predicted motif plots were generated with 100,000 9-mer peptides randomly sampled from the human proteome and selecting the top 1000 by HLApollo score. Observed motif plots were generated with 756, 2087, and 1367 9-mer peptides for B*14:01, B*57:02, and B*51:05, respectively. AP numbers were generated with 986, 3755, 2130 positive peptides and 99:1 negative:positive ratio. Source data are provided as fig7-b_B*14:01_true.csv, fig7b_B*14:01_pred.csv, fig7b_B*51:05_true.csv, fig7-b_B*51:05_pred.csv, fig7b_B*57:02_true.csv, fig7b_B*57:02_pred.csv.

data is needed to calculate those features). To this end we investigate a feature which is available a priori, the MHC NN BLOSUM distance (Fig. 8c), which only relies on MHC information and not ligand information. One may observe modest correlation between the drop in trained vs untrained AP (AP drop) for this feature (Fig. 8c). We observed (Fig. 8d) the untrained motifs based on the model (upper panels) and observed (lower panels) motifs for the alleles with the closest (left panels) and furthest (right panels) MHC NN BLOSUM distance. Good agreement for the close allele's untrained motif and the observed motif is observed, but the distant allele's untrained motif underrates the importance of the unusual anchor residue at position 5. Taken collectively, MHC NN BLOSUM distance is a useful a priori feature that correlates with the AP of HLApollo on untrained alleles.

Next, we propose a clinically useful, a priori model for predicting untrained AP (Fig. 8e) with MHC NN BLOSUM distance and several other parameters (depicted in the lower half of Fig. 8f). Most of these parameters are similar to those in the a posteriori model, but using (untrained) model predictions on random peptides from the human proteome to extract the parameters. This model achieves reasonable predictive value (Pearson's correlation of 67%), and so we seek to use it to evaluate if HLApollo predictions should be considered useful for a particular untrained allele. To be considered 'covered' by HLApollo we demand a threshold of predicted untrained AP of at least that of the worst performer amongst the trained alleles, 67.8%, plus one standard deviation of error in the predictive model, a total of 76.7%. For a comprehensive estimate of genotype coverage across ancestries we queried http://allelefrequencies.net for allele frequencies from only gold-standard datasets with at least 1000 samples (data collected October 6 2021) and evaluated coverage on the 2627 alleles identified[48]. Generalization accounts for HLApollo coverage on 791 alleles, with particularly large gains in genotype coverage amongst several ancestries, including Asian and Hispanic, shown in Table 1. Thus, based on predicted generalization, HLApollo's coverage extends substantially beyond the alleles in its training data, especially for ancestries underrepresented in the training data.

Discussion

With the recent surge in the availability of large-scale immunopeptidomics data, several deep learning approaches have been applied to the problem of predicting peptide presentation by MHC-I, including convolutional neural networks[34], feed forward neural networks[28], mixture models[40], LSTM[49], etc. The inductive bias of sequence-based neural networks like transformers and recurrent neural networks is well suited for the problem, as distant amino acids can impact one another's representation. In this study, we implement a highly accurate, transformer-based approach for pan-allelic peptide presentation prediction by MHC-I, HLApollo. Our approach provides a more generalized representation of the peptide presentation problem compared to heuristics usually implemented in other methods. We achieve this by using a pan-allele approach and performing inherent deconvolution when multiallelic data is input. This is possible through our use of a dummy beginning-of-sequence (BOS) embedding, which enables the model to transfer information from the peptide and MHC sequences to a fixed size vector embedding, where the information transfer is enforced by exclusively using the BOS embedding for classification. We also use a negative set switching approach that increases the diversity of the negative space observed during training without overfitting to it. This proved to be important for improving performance, enabling a large model that can capture pan-allelic generalization. This issue arises due to the unlabeled nature of the training data, as ligandome data sets do not have experimentally labeled negatives. The NNAlign_MA[39] method uses a data partitioning approach which likely addresses this issue as well, but we believe several, if not all, other contemporary models do not address this issue. Our *bonafide* deconvolution strategy also enables the best use of multi-allelic data to date, achieving the smallest gap between SA-only and MA-only performance on BM1 of the models we evaluated,



Coef	SE	Prob	Parameter
0.7475	0.009	0.000e+00	Intercept
-0.0749	0.011	1.200e-09	Peptide NN Blosum Distance
-0.0530	0.010	3.228e-07	Training Count
-0.0769	0.013	8.350e-08	Motif Entropy
0.0486	0.012	7.265e-05	# Presenters
-0.0743	0.014	5.859e-07	Motif Abundance

Coef	SE	Prob	Parameter
0.7475	0.012	0.000000	Intercept
-0.0156	0.017	0.355314	Peptide Blosum NN Distance
0.0190	0.016	0.230814	Peptide Blosum NN AP
-0.0245	0.015	0.110312	MHC Blosum NN Distance
0.0152	0.015	0.326039	MHC Blosum NN AP
-0.0614	0.021	0.005352	Motif Entropy
0.0603	0.017	0.000444	# Presenters
-0.0583	0.020	0.005214	Motif Abundance

Fig. 8 | Performance on untrained alleles is predictable. a depicts a scatter plot of trained vs untrained AP for all the single allelic datasets used in training, bubble size indicates number of peptides from the corresponding allele. There are 31, 50, and 17 HLA-A, -B, and -C alleles, respectively. The inset depicts a histogram of the AP drop from full HLApollo (**b**) depicts the correlation between the actual untrained AP, and the predicted untrained AP by the a posteriori untrained AP multilinear regression model. Pearson's correlation and p-value (2 sided) were calculated with SciPy stats. **c** depicts the AP drop between trained and untrained models vs MHC NN BLOSUM distance for HLApollo. **d** depicts the observed (lower) and untrained HLApollo predicted (upper) motifs for A*02:01 (left, 4136 peptides to generate data motif) and B*37:01 (right, 548 peptides used to generate data motif). Predicted motif plots were generated with 100,000 9-mer peptides randomly sampled from the human proteome and selecting the top 1000 by HLApollo score. **e** depicts the correlation between the actual untrained AP, and the predicted untrained AP by the a priori untrained AP multilinear regression model. **f** depicts the coefficients, standard errors and probabilities of the parameters used in the a posteriori (upper) and a priori (lower) untrained AP models. *p* values calculated from t-test with two sides. Standard error (SE), probability (prob), Average Precision (AP). Source data are provided as fig8a.csv, fig8b.csv, fig8c.csv, fig8d_A*02:01_pred.csv, fig8-d_B*37:01_pred.csv, BM1.tar.gz, fig8d_pred.csv, fig8e.csv.

indicating superior deconvolution capability. Unlike the recent transformer approach, TransPHLA, we do not concatenate all token representations and send them through fully connected layers, as this reintroduces issues with padding (e.g. representation of padded tokens are processed by the model) and leaves only 1% of model parameters in the transformer encoder (which we found led to poor performance on our test dataset).

Our approach does have some limitations. As a transformer model with multiple layers, the model has a significant number of parameters. Many of these parameters are used to embed the MHC pseudosequence for which there are a limited number of examples. This could cause overfitting (although good MHC generalization was still observed for our model). In future work, better strategies to embed MHC representations could be developed.

While other approaches normalize the ranking of different alleles by calculating the % rank of a given peptide with respect to each allele, we opt against this strategy. From Supplementary Fig. 5, one may observe significantly lower HLApollo logit scores for alleles from HLA-C than HLA-A and HLA-B, we believe this to represent biological differences in the presentation rates between genes, and so avoid normalization of scores for different alleles. Further, we do not use binding affinity data, which is allele-specific as a different reference peptide is used for each allele in binding assays (which measure affinity based on exchange of the reference peptide at increasing concentrations of query peptides). Elution assays do not use such reference peptides, precluding the need for normalization. We also constructed a single pan-allele model instead of multiple allele-specific models, as done in HLAthena. Taken together, a calibration step is not needed in our approach, thus avoiding yet another heuristic. However, we note that inherent allelic biases because of antibody pulldowns might still be present, especially in MA data.

Both NetMHCpan4.1-EL and our ensembled feed forward neural network fell significantly short of HLApollo (Fig. 2) demonstrating the importance of using a sequence-based neural network over ad hoc anchor placement conserving padding strategies. This was shown despite important biases that favor NetMHCpan4.1-EL. 34.9 % of our positive BM1 test set evaluation data appears in NetMHCpan4.1-EL's training data (MixMHCpred2.2 is also trained on 43.24% of our test set). NetMHCpan4.1-EL and NetMHCpan4.1-BA (NetMHCpan4.1 BA is the p-MHCI binding affinity predicted by the NetMHCpan4.1 model) have been used pervasively in selecting epitopes for immunogenicity studies, which can potentially bias results in its favor[27]. Additionally, peptides chosen for BA datasets are often enriched for T-cell epitopes, and many studies investigating T cell response use a binding affinity model to select peptides[29].

Besides modeling peptide presentation purely based on amino acid sequences, we also considered other contributing factors, like expression of the source gene and sequence properties of the source protein. Several studies have explored and demonstrated the benefit of using expression information for the task of predicting peptide presentation. The sources of expression information in these studies include the sample-specific expression values, or a reference RNA-seq data set of matched tissue[42,50], and some studies have used protein abundance instead of gene expression[51]. Inclusion of expression

Table 1 | Projected generalized coverage of HLApollo for several ancestries

Ancestry	Trained Genotype Coverage	Generalized Genotype Coverage
Amerindian	82.9%	86.5%
Arab	86.9%	87.4%
Asian	79.7%	84.2%
Austronesian	47.6%	67.0%
African	89.2%	90.6%
European	93.8%	95.1%
Hispanic	70.0%	80.5%
Mestizo	47.7%	66.8%
Polynesian	50.1%	70.5%

The full genotype coverage (HLA-A1 * HLA-A2 * HLA-B1 * HLA-B2 * HLA-C1 * HLA-C2, i.e. all 6 alleles are available in the training data or are expected to pass a threshold AP value of 76.7% for HLApollo) for various ancestries. The center column shows the coverage using only alleles found in our training dataset. The right column shows the coverage when pan-allelic generalization is accounted for. Source data are provided as "Table 1.csv".

information boosted accuracy, primarily by attenuating sequence-based prediction at low expression levels. The benefit of using expression information is likely to be retained even when using proxy expression features from reference data sets, but exploring this is out of scope of this study. We argue that protein compartmentalization and other features of source proteins that affect their propensity for peptide presentation are encoded in the protein sequence, and develop a gene-level propensity score, $\mathrm{ESM}_{\mathrm{MHC\text{-}I}}$, by training a model which maps protein language representations to presentation likelihood. $\mathrm{ESM}_{\mathrm{MHC\text{-}I}}$ also boosts the accuracy of HLApollo, and to a better extent than including manual annotations like Gene Ontology. Surprisingly, the boosts by expression and $\mathrm{ESM}_{\mathrm{MHC\text{-}I}}$ were not synergistic, indicating that $\mathrm{ESM}_{\mathrm{MHC\text{-}I}}$ might have intrinsically encoded some expression biases by protein families. For instance, transcription factors are generally expressed at low levels. Our findings also suggest that HLApollo + $\mathrm{ESM}_{\mathrm{MHC\text{-}I}}$ can be useful in applications where expression information is limited. This method showed performance gains in tissue hold-out presentation data, and also in T cell response data (Figs. 4/5). However, when actual expression information is available, HLApollo + Expr shows superior performance (Fig. 5b).

Lastly, pan-allelic modeling is critical for improving patient inclusion in clinical trials, and so it is a primary goal of this work to investigate the generalization of HLApollo to new studies and new alleles. The linear regression model in Fig. 8e enables the prediction of model performance on untrained alleles using only a priori knowledge gathered from either the allele pseudosequence or the model's preconceived notions of the allele. Using confident generalizers benefits people of color, who are unfortunately underrepresented in currently available ligandome data. This method also enables the selection of alleles for acquiring ligandomes by prioritizing predicted low performers with large population frequencies.

Methods

Generation of single-allelic cell lines, cell culture and expansion

Generation of the HMy2.CIR (C1R) HLA class I knockout parental cell line is described previously[52]. Cells were maintained in IMDM +Glutamine media (Gibco) with 10% FBS (R&D Systems).

Stable mono-allelic cell lines were generated through transduction of the parental line with lentiviral constructs expressing the HLA allele of interest. Lentiviral HLA expression constructs and lentiviral packaging plasmids were transiently co-transfected at a molar ratio 1.5:1.35:0.345 (pHLA:Delta8.9:VSVg) into 293 T cells with Lipofectamine 2000 (Invitrogen) following the manufacturers recommendations. Lentiviral supernatants were harvested at 72 hours and filtered through a 0.45 μm PES syringe filter (Millipore). Viral concentration was performed using LentiX Concentrator (Takara bio) using the manufacturer's protocol. Concentrated virus was stored at −80 °C for future use.

For viral transduction of HLA allele expressing lentivirus, 1 million parental C1R knockout cells were plated in 500 μl of media with 8 μg/ml polybrene (Millipore-Sigma). Cells were centrifuged at room temperature for 30 min at 800 × *g*. Following transduction, 500 μl fresh media was added and cells were incubated overnight. The following day, viral-containing media was removed. Cells were subsequently expanded for a minimum of 72 h.

Bead-based HLA enrichment of transduced cell populations

Streptavidin magnetic beads (MACS) were use to enrich for HLA positive cells from infected populations. 10 million cells were washed and resuspended in 100 μl PBS, 2 mM EDTA and incubated with 5 μl pan-HLA-biotin antibody (Biotin anti-human HLA-A, B, C, clone W6/32 mouse IgG2a, Biolegend #3) for 30 minutes at 4 °C. Cells were washed twice with MACS buffer before being processed via MS columns (Miltenyi) following the manufacturer's recommendations. Eluted cells were subsequently passed over a second MS column to further ensure highly pure cell populations (Supplemental Fig. 12).

Flow cytometry

Following cell expansion HLA expression was evaluated by flow cytometry (Supplementary Fig. 12). 1 million cells were washed one with FACS buffer and resuspended in 95ul FACS buffer and 5ul primary antibody (APC anti-human HLA-A, B, C, clone W6/32 mouse IgG2a, Biolegend #3) and 1ul of VioBlue-Viobility dye (Miltenyi). Cells were incubated at 4 degrees for 20 minutes then washed twice with FACS buffer before analysis.

Cell expansion and sample preparation

Once the cell number reached near 200 million cells, each cell line was passaged into a 1 liter Spinner Flask with 600-750 ml fresh medium. When the cells reached near 1 million cells/ml, the HLA class I expression was assessed by flow cytometry as described above immediately before cell pellet preparation. One large cell pellet of 500 million cells was washed in PBS and flash frozen utilizing a dry-ice bath and then stored at −80 °C. Cell expansion and subsequent processing of some samples were performed by Cayman Bio.

pMHC-I immunopurification, LC-MS/MS

Immunoprecipitation was performed on 500 × 10^6 cells for each sample. The cells were lysed in 0.25 % Sodium deoxycholate, 200 μM iodoacetamide, 1% N-Octyl-β-D-thioglucoside, 1 mM EDTA,1 protease inhibitor tablet per 10 mL of DPBS buffer and immunoprecipitated with pan anti-HLA-A, B, C antibody purified from hybridoma (W6/32 clone) cell line (H, ATCC) that was immobilized and covalently linked to Protein A cartridges (Agilent G5496-60000). Peptides were acid eluted from antibody bound Protein A cartridges using 0.1 M acetic acid/0.1%TFA followed by desalting with C18 solid-phase extraction (SPE). The eluted peptides were injected onto an analytical column packed with Luna C18 5 μm 100Å resin (04A-4252, Phenomenex) into 75um × 25 cm picofrit column (PF360-75-15-N-5, New Objective) and separated at a flow rate of 350nL/min over a linear gradient from 5% to 25% buffer B (acetonitrile in 0.1% formic acid) for 95 min, 25% to 35% B for 15 min, 35% to 90% buffer B for 2 min, and held at 90% B for 2 min. The LC eluent was directed into an Orbitrap Fusion Lumos Tribrid mass spectrometer (Thermo Fisher Scientific) equipped with a nano-electrospray ionization source with spray voltage set at 1900 V. Mass spectral data were acquired using Orbitrap MS scans (R = 60,000 at m/z 400) at an AGC target value of 4 × 10^5 with a max IT of 50 ms. The full MS scan covered a range of 300–700 m/z. MS2 scans were acquired in the Orbitrap (R = 15,000 at m/z 400) in data dependent mode of 3 s cycle time, scan range set as Auto Normal, dynamic exclusion set at 25 ms, AGC target at 1E5, and maxIT of 250 ms. Peptide precursors were subjected to CID with NCE@35% and EThCD fragmentation.

MS data analysis and peptide identification

The tandem mass spectral data were searched using PEAKS Studio (v10.5)[53] against the Swiss-Prot human database (downloaded 01 Jan 2020) with no enzyme specificity, peptide mass tolerance at 10ppm, fragment mass tolerance at 0.02 Da, variable modifications on Oxidation (M) and Deamidation (NQ). Search results were filtered to an estimated peptide false discovery rate of 1%.

Ligandome data processing

Positive peptides originally derive from searching spectra generated by liquid chromatography–tandem mass spectrometry experiments on eluted MHC ligands. Data was collected from 23 published studies (Supplementary Data 1) along with IEDB (originally from[34]) and deposited in a custom sqlite database, mhcDB.

IEDB data was initially downloaded from:

https://iedb.org/downloader.php?file_name=doc/mhc_ligand_full_single_file.zip on March 19, 2019. Only peptide/allele pairs collected via mass spectrometry were used for training and evaluation of HLApollo. More specifically, this corresponds to all rows containing the string, "mass spectrometry", in the "Method/Technique" column. Next, all rows corresponding to the studies in Supplementary Data 1 were removed. This was achieved by first removing rows where the "PubMed ID" values were equal to the "PMID"s in Supplementary Data 1. Additionally, rows corresponding to Schuster et al.[44], determined by finding rows where the "Authors" column contained the string, "belin". In total, rows corresponding to 15 of the 22 published studies were removed. Additionally, only rows containing wildtype HLA-A, HLA-B, or HLA-C alleles at 4-digit resolution were retained via regex on the "'Allele Name" column. Lastly, peptides containing any non-canonical amino acid residue or post-translational modification were removed.

For training and evaluating presentation, we filtered a total of 142,298 positive peptides for a variety of reasons. Peptides were mapped to translated transcripts from Gencode 27, Basic set (GRCh38.p10)[54] for the purposes of unambiguously modeling cleavage and source gene mRNA expression. Peptides that mapped to multiple genes were dropped from training and test sets, to enable unambiguous training of peptide presentation based on gene expression (54,056 peptides). Similarly, peptides that mapped to multiple isoforms of the same gene were dropped if they had different flanking sequences of 10 amino acids, to enable unambiguous encoding of peptide processing information (42,540 peptides) (Fig. 1c). Moreover, peptides that did not unambiguously map to the human proteome were discarded from presentation benchmark sets (36,553 peptides). Additionally, we did not consider peptides with post-translational modifications (24,341 peptides) or that fell outside the length range of 8-14 amino acids (29,566 peptides). An additional 21,816 positive peptides were excluded from presentation training and evaluation as these were segregated for future internal use as a test set.

Since experimental negative peptides cannot be obtained from current approaches, we use the non-presented parts of the human proteome to sample negative peptides. Notably, a small fraction of these computationally identified negatives might be presentable, but were not identified in the current ligandome data. These are 'gold std false negatives', and presumably they constitute a small fraction that will not affect machine learning significantly. First, we sampled negative peptides without replacement, per genotype, across the human proteome with a uniform probability of peptide lengths, 8–14. For each single-allelic genotype, positive peptides from other multiallelic genotypes with a matching pseudosequence were excluded from the negative set. For each multi-allelic genotype, positive peptides from matching single-allelic genotypes with a matching pseudosequence were also excluded from the negative set.

BM1, BM3, Sarkizova Holdout, BM7, K-folds, BM4, BM5

For BM1 (presentation evaluation) the positive elution benchmark dataset was first constructed, per genotype, by randomly selecting 90% of available {peptide,genotype} tuples for training and 10% for testing after filtration (see above). Next, the negative training benchmark dataset was constructed, per genotype, at a 1:1 negative:positive ratio for 520 mutually exclusive sets of {peptide,flank,genotype} tuples in the training split and 4999:1 negative:positive ratio for a single set in the test split. We subsetted this to a ratio of 99:1 for evaluation of HLApollo (Fig. 2), and kept the 4999:1 ratio for evaluation of HLApollo + Expression (Fig. 3). Finally, to eliminate any possible test set leakage, we enumerated all possible {el,peptide,pseudosequence} tuples in the train and the test splits, took their intersection, and then dropped, from the test data, any {el,peptide,genotype} tuples that mapped to this intersection. Note that 'el' refers to the elution outcome (value of 0 or 1). Thus, any pair of peptide and MHC(s) in the training data is prevented from leaking into the test data. An alternative test set where all peptides are dropped between test and train is also evaluated, with results shown in Supplementary Fig. 8.

The BM3 training dataset for evaluating tissue holdouts was constructed from BM1 by removing any genotypes that matched those derived from the Schuster, Loffler, or Pyke studies according to our mhcDB. Then the dataset was rebalanced at a 1:1 negative:positive ratio per genotype. The BM3 test dataset was taken directly from Supplementary Data 1c from[27]. To account for missing flanking sequences from this test set, peptides were remapped to the human proteome (GRCh38.p10) and flanks of length 10 and 30 were used to predict HLApollo and HLAthena MSiC, respectively. In cases where a single peptide mapped to multiple flanking sequences, HLApollo and HLAthena predictions were averaged. In cases where a single peptide did not map to the proteome, flanks were not considered in the HLApollo prediction and HLAthena MSI score was used. Finally, to eliminate any possible test set leakage, we enumerated all possible {el,peptide,pseudosequence} tuples in the train and the test splits, took their intersection, and then dropped any {el,peptide,genotype} tuples from the test set that mapped to this intersection.

The Sarkizova Holdout training set was constructed by excluding all {peptide,genotype} tuples derived from the Sarkizova study from the BM1 training dataset. Similarly to BM1, we maintained a 1:1 negative:positive ratio in the training split across 520 sets of negative {peptide,flank,genotype} tuples. The Sarkizova Holdout test set was generated as the subset of the BM1 test set derived from the Sarkizova study and a 99:1 ratio was maintained in the test set. The Sarkizova Holdout MA model was trained only on the MA subset of the Sarkizova Holdout training set.

The K-folds datasets were generated similarly to BM1. First, the rows of the BM1 positive data were randomly shuffled. Next, the shuffled dataset was evenly split into 5 sets. For each of the 5 sets, 1 was used as a test set and the remaining 4 were used for training. For each of the 5 sets, negative train and test data was generated similarly to BM1. For each of the 5 sets, test set leakage was removed similarly to BM1, except using 50 negative ensembles instead of 520.

The cancer neoantigen datasets, BM4 and BM5, were collected from the supplementary data from their respective publications[26,29]. Flanking sequences were obtained by mapping the wildtype peptide to the human proteome (GRCh38.p10).

BM6 (Deconvolution simulation)

While controlling for peptide count and pairwise motif similarity, we restricted half of the alleles to belong to unique combinations of synthetic samples. We term these as 'deconvolvable' alleles, as presumably their motifs would be expected to be present in the unique combination of samples, hence identifying them. The other set of alleles were assigned to non-unique sets of synthetic samples (hence termed 'non-deconvolvable alleles')

Alleles with at least 1000 unique peptides (72 alleles) were first selected to mitigate performance bias due to lack of training data. To ensure even distribution of peptide counts across alleles, each allele contributes exactly 500 randomly selected peptides to a synthetic sample. The SA training data was constructed from this dataset, containing 140,530 unique {allele, peptide} tuples.

For MA training data, similar to the biological scenario, a minimum of 1 and maximum of 2 alleles per gene (A, B, C) were necessary to define a synthetic sample. To attenuate the potential impact of pairwise allele similarity on deconvolution performance, pairwise allele similarities informed constraints on which pairs of alleles were allowed to belong together in synthetic samples. Specifically, distance was defined as Jensen Shannon Divergence between 9-mer peptides in the training data of the selected alleles and a threshold of 1.23 was determined as the minimum distance allowed between two alleles (based on visual inspection of motifs). Any peptides that were duplicated per sample were removed.

To explore model performance in the context of principles of deconvolvability of allele-specific peptides based on co-occurrence and exclusion of alleles across samples, roughly half (36/72 alleles) were randomly forced to always occur in pairs when assigned to synthetic samples ('non-deconvolvable'). The remaining alleles were assigned randomly to synthetic samples, each occurring in a unique set ('deconvolvable').

To ensure a dataset of realistic size, 195 synthetic samples were generated, yielding 514,029 total unique genotype-peptide tuples encompassing 107,362 unique peptides.

To ensure no overlap of positive and negative peptides when combining alleles, any overlapping positive/negative SA peptides were removed from consideration of the negative peptides for a given synthetic sample.

We define the 4 categories of deconvolution difficulty as follows: Trivial deconvolvable alleles found in MA genotypes are those for which SA data exists in our dataset. Easy deconvolvable alleles are those for which SA data is absent, but all other alleles within the genotype in MA data are trivially deconvolvable, thus effectively making the remaining allele deconvolvable. Medium deconvolvable alleles are those for which all other alleles within the genotype are either trivial or easy deconvolvable alleles, leaving the medium allele to be deconvolvable. Hard deconvolvable alleles are those for which no combination of trivial, easy, medium (or iterations beyond medium, where medium alleles are used to deconvolve remaining alleles) alleles can isolate the allele. For comparison with the co-occurence and exclusion definitions of deconvolvability, both medium and hard deconvolvable alleles are considered 'non-deconvolvable'.

Gene expression

Fastq files were collected from bulk RNAseq expression profiling studies on specific ligandome samples (e.g. Shraibman_MA_2016), participants (e.g. cell lines, Schuster_ovarian_2017), and representative healthy tissue atlas specimens. First, RNAseq sample runs were aligned using HTSeqGenie[55,56].

Sample runs with fewer than 10 million unique/concordant mapped reads were discarded from downstream analysis. Other QC metrics (e.g. percent mapped ribosomal reads, mitochondrial reads, etc.) were visually inspected but no sample runs exceeded recommended QC cutoffs. After initial upstream QC, samples were pseudo-mapped / quantified using Salmon (v1.3.0)[57] on the Ensembl90 reference transcriptome. This approach allowed us to increase sensitivity via consideration of multi-mapped reads and quantify transcript expression as a function of sequence-specific bias, thereby attenuating potential batch effects.

We also considered such a heterogeneous dataset may contain batch effects due to various technical biases such as paired-end vs. single-end sequencing, strand protocol, read length, and sequencing instrument. We conducted principal component analysis to examine such biases. No principal components obviously separated technical features (shown by visual inspection, binomial/multinomial regression of principal components onto features).

We matched RNAseq expression to immune peptidomics data in BM1 by joining RNAseq assay data (Database table mhcDB::Assay_rnaseq) on one of either: mhcDB::Sample.id, mhcDB::Subject.id, or mhcDB::Tissue_atlas.id - depending on the granularity available. We define the entity RNAseq sample as a unique combination of: mhcDB::Sample.id, mhcDB::Subject.id, mhcDB::Tissue_atlas.id, mhcDB::Genotype_mhc_I_human.id.

We quantified gene expression for a sample as the sum of transcript TPM per RNAseq assay from that sample. Then we averaged gene expression across all the different RNAseq assays, if multiple assays were available, for a given RNAseq sample (defined above).

We construct our expression evaluation dataset independently for each RNAseq sample, and in each case, restrict the universe of possible peptides (+ve and -ve) to 50,000 total peptides. Hence, the ratio of negative to positive peptides may vary between RNAseq samples, but the proportion of positives can be interpreted as the probability of peptide presentation for a given RNAseq sample. The empirical enrichment score for peptide presentation is described in supplementary methods (section Gene expression).

GO gene sets

We used gene sets that correspond to the cellular component (CC) terms from Gene Ontology (GO)[58]. Specifically, we selected GO CC terms that consist of major subcellular localization based on manual interpretations, and cellular components that have been shown to be associated with MHC-restricted peptides[14,59], namely: nucleus (GO ID, GO:0005634), mitochondrion (GO:0005739), proteasome complex (GO:0000502), endoplasmic reticulum (GO:0005783), Golgi apparatus (GO:0005794), cytosol (GO:0005829), cytoskeleton (GO:0005856), cytoplasm (GO:GO:0005737), plasma membrane (GO:0005886), cell junction (GO:0030054), extracellular region (GO:0005576), lysosome (GO:0005764), endosome (GO:0005768), early endosome (GO:0005769), late endosome (GO:0005770), endocytic vesicle (GO:0030139), cytoplasmic vesicle (GO:0031410), vesicle (GO:0031982), recycling endosome (GO:0055037), and secretory vesicle (GO:0099503). Gene annotations found associated with the respective GO CC terms were then obtained from Ensembl release 90[60].

Motif logos

Positive single-allelic peptides and their (N, C)-terminal flanking sequences were taken from BM1. The background distribution of amino acids and of terminal ends of proteins were computed from the proportions found in the negative peptides (and their flanking sequences) from BM1 (ensembles 1-5, note that the negatives are genotype-specific, so anchor preferences for different alleles get diluted over the space of all negatives). A matrix P_{ai} was computed and used as input to R's ggseqlogo package or Python's logomaker package[61,62] to plot the motif logo. The Kullback-Leibler divergence calculation performed for all positions and amino acids is described in the supplementary "methods" (section Motif logos).

Model architectures and training

All artificial neural network models in this work are implemented in pytorch[63], and training loops in fastai[64], on (which includes features such as the default fit one cycle learning rate scheduling) with mixed precision. Binary cross entropy loss and the Adam optimizer[65] are used in their default settings. HLApollo uses a batch size of 3072 and a learning rate of 0.0002, except for the final transformer encoder layers and fully connected layers, which use a learning rate of 0.0004. Weights are initialized with default Xavier initialization. Single allelic (SA) samples are upweighted in the loss function by a factor of 4. The feed forward neural network (FFNN) uses a learning rate of 0.0006, and batch size of 900. HLApollo is trained for 115 epochs, and FFNN is trained for 50 epochs. HLApollo's training time on one machine with 8 NVIDIA V100s is 7.7 hours per model. Ensembling is performed by averaging the predictions from each model, with 10 models. The datasets are subsampled to 80% each epoch to increase diversity across ensembles. Manual hyperparameter optimization is performed for each model, due to the training time involved for each model. Negative set switching is implemented by sampling a new negative set from a space of 500 full negative sets (fewer for those depicted in Fig. 2b) per epoch.

HLApollo's architecture is as follows. Amino acid tokens are embedded by a learned embedding (pytorch's nn.embedding) to a dimension of 400. Flanking sequences that reach the edge of the protein are given a special token (this token is padded to the length 10 if relevant). As described in the main text, flanking sequences are randomly dropped 50% of the time, and 50% of the time 0-10 amino acids are removed (following a uniform distribution). The n-flanking, peptide, and c-flanking sequences are padded to their maximum lengths (10, 14, and 10 respectively) and concatenated together (now termed the concatenated sequence) and standard transformer positional encoding is performed. The MHC sequences are treated with a separate embedder and positional encoder (same implementation as for the peptide sequence); all layers that process MHC sequences have shared weights for the various MHC sequences in the multiallelic (MA) setting. The concatenated sequence (peptide and flanks), and its padding mask (to prevent unused tokens from impacting other token's representations), are sent through a transformer encoder layer. All transformer encoder layers used have a dimension of 400, 16 heads, and no dropout. The MHC sequence is separately passed through its own transformer encoder layer. The concatenated sequence and MHC sequence are then further concatenated, for each MHC sequence, and a beginning of sequence (BOS) token (passed through its own embedder and positional encoder) is concatenated to the beginning of each sequence, called the pair sequence. Without gradient accumulation, all pair sequences are passed through the output module: 4 transformer encoder layers. Next, the BOS token is fed through, sequentially, a fully connected (FC) layer (dimension 256), a dropout layer (50%), a swish activation[66], a second FC layer (dimension 128), a second dropout layer (50%), a second swish activation, and finally a FC layer to output a single logit score. Used in this way, the BOS token causes information from the rest of the sequence to flow into the token, creating a sequence representation with a fixed size, regardless of the sequence length. Finally, the concatenated sequence-MHC pair with the highest logit score is identified, and this pair is run through the output module with gradient accumulation (this is our deconvolution strategy). Removing the gradient accumulation for pairs that are not determined by the model to be the presenting pair is used to prevent gradient noise (and increase speed) that would accumulate if each is allowed to pass through the output with gradient accumulation.

We performed a rough, manual, hyperparameter sweep over learning rate (lr), model dimension, head dropout, and transformer dropout. BM1 performance was used to select the hyperparameters. Batch size was fixed at 3072 to maximize the utilization of a V100 GPU, lr was tested between 0.0001 and 0.01 with 3 rates for each power of 10. Model dimension was varied between {60,120,200,400,600}, with the head dimension constant. Head and transformer dropout were carried between {0,0.2,0.4,0.5,0.6}. Little impact on model performance was found for most hyperparameter values (other than lr).

The FFNN uses the Netmhcpan[20] inspired 9-mer mapping commonly used in the literature. 8mers were increased to a length of 9 by adding an X token in between two existing amino acids, resulting in 7 enumerated peptide possibilities. Peptides longer than 9 had (length −9) consecutive amino acids removed from their sequence, resulting in 9 peptide possibilities. N-flanking and c-flanking sequences were padded with X tokens to the maximum flanking sequence length (10). N-flanking sequences, possible peptide sequences, c-flanking sequences, and MHC sequences were concatenated, for each peptide possibility and each MHC sequence (for example a 10-mer peptide for a 5-allele sample will have 45 sequences associated with it). These paired sequences were BLOSUM62 encoded[67], and sent into the FFNN model (input size 1050), without gradient accumulation. The FFNN model is composed of the following sequential layers: a FC layer (dimension 512), ReLU activation, batch normalization, a second FC layer (dimension 512), a second ReLU activation, a second batch normalization, and a final FC layer to output to a logit score. The peptide possibility - MHC pair with the largest logit score was then chosen for each input and sent through the model again with gradient accumulation.

The non-pan allelic HLApollo was trained with a similar strategy to MHCNuggets[49], the largest SA dataset (B*27:05) was used to train HLApollo as described above, then transfer learning was used (learning rate 10% as above) to train the other SA models.

ESM1b protein features were extracted by taking token representations of amino acid residues in a particular protein following the details described in their github (https://github.com/facebookresearch/esm). To produce a protein-level feature set we averaged the residue features across the protein, resulting in a 1280 dimensional vector. The $\text{ESM}_{\text{MHC-I}}$ model is a FFNN model that takes the 1280 dimensional protein vector from ESM1b and maps it to a presentation likelihood using two FC layers (500 and a 50 dimensional), with dropout (50%) applied after each hidden layer. We used a batch size of 1024 and a learning rate of 0.001. The model was trained by taking each peptide from our baseline training dataset, determining the protein from which it was derived, and using its protein features as input.

Logistic regression models used to add expression, gene ontology (GO), and $\text{ESM}_{\text{MHC-I}}$ features to HLApollo score were implemented in sklearn using default settings, except the maximum iterations allowed were set to 3000. Pseudo counts were added to the TPM from the source gene and it was log transformed.

We calculated the % rank by acquiring predictions on random peptides sampled from the human proteome of length 8-14 paired with alleles in our dataset and mapping the logit score to the % rank on this dataset. This is done across all alleles, as opposed to an allele-specific normalization approach. The dataset's allele distribution is chosen to match that of our test dataset, and the peptide length distribution is uniform.

We evaluate our models primarily using the average precision metric. Average precision (AP) is calculated with the python package sci-kit learn, and is the AP over all threshold values that change the number of predicted positives. This is equivalent to the area under the curve of a precision-recall curve (for this reason its maximal value is 1 and so we report AP as a percentage). We choose to use this metric over ROC-AUC, which tends to give very high performance values when the proportion of test peptides is highly skewed towards negatives, and is more appropriate in the case of a balanced test set or when the number of test data points is small.

Pan-allelic generalization methods

Each allele out of training (OOT) model is trained by removing an allele with SA data, and all MA genotypes that the allele appears in from the training dataset, this is performed for each allele with SA data. Training is performed as described above.

The prior knowledge linear regression model, implemented in sklearn using default settings, is trained using the following (scaled) parameters: 1) The peptide BLOSUM distance to its nearest neighbor (NN) in the training dataset. Amino acids are embedded in the BLOSUM62 space, and averaged position wise across each allele's peptides, and then these 9 vectors are concatenated to form one vector for each SA allele, thus representing the 'average' ligand for each allele. For each held-out allele, the training allele with the minimal distance in this space is found and this distance is used as the peptide BLOSUM distance to its NN. 2) The number of peptides in the SA dataset of interest. 3) The peptide entropy of interest. This is calculated by generating a motif for each 9-mer peptide of interest (implemented with logomaker[61]), and summing the entropy over the motif. 4) The # of peptides with HLApollo logit score over 0 on a dataset of 100,000 randomly sampled peptides from the human proteome. 5) The motif abundance, proposed in Sarkizova et al. [28] which sums the frequency of each amino acid in a 9-mer motif normalized by the frequencies of the amino acid occuring in the human proteome.

The a priori linear regression model, also implemented in sklearn using default settings, is trained with the following (centered and scaled) parameters, which have all been obtained by getting the appropriate OOT HLApollo predictions on 100,000 peptides randomly sampled from the human proteome, and considering those with logit score over 0 to be likely presenters: 1) The peptide BLOSUM distance to its NN in the training dataset. 2) The AP of the model on this peptide-based NN dataset. 3) The MHC BLOSUM distance to its nearest neighbor. 4) The AP of the model on this MHC-based NN dataset. 5) The predicted motif entropy. 6) the predicted # of presenters out of 100,000. 7) The predicted motif abundance. Inputs to both the prior knowledge and a priori linear regression models are centered and scaled prior to fitting.

Allele and genotype coverage is assessed by querying allele-frequencies.net for allele frequencies from only gold-standard datasets with at least 1000 samples (October 6 2021) and evaluated coverage on the 2627 alleles identified[48]. Allele coverage is found per ancestry and is converted to allele coverage by the following equation:

$$\text{Genotype Coverage} = \text{HLA} - \text{A Coverage} \times \text{HLA} - \text{A Coverage} \times \text{HLA} - \text{B Coverage} \times \text{HLA} - \text{B Coverage} \times \text{HLA} - \text{C Coverage} \times \text{HLA} - \text{C Coverage}$$

Untrained alleles are considered covered by HLApollo if the a priori model determines that the AP is one standard deviation above the worst performer amongst the trained alleles, 67.8%, a total of 76.7%.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All datasets utilized in this study, including those used for benchmarking, the mhcDB sqlite3 database, and a merged table derived from the mhcDB, are comprehensively provided, as well as all source data used to create the figures at https://doi.org/10.5281/zenodo.7951717. Note, the training and test sets are in version2 of this repository that can be accessed at the same link as above. Researchers and interested parties are encouraged to explore and utilize these datasets in accordance with the guidelines provided in the repository. All other data are available in the article and its Supplementary files or from the corresponding author upon request. Source data are provided with this paper.

Code availability

Downloading, installing, or using HLA Apollo is subject to agreement to the terms of the HLA Apollo License attached herewith. The executable is available at https://github.com/Genentech/HLApollo and at Code Ocean DOI: https://doi.org/10.24433/CO.4653665.v1. The latter also contains a readily executable example. Please note that the HLA Apollo licenses allows HLA Apollo to be downloaded, installed, and/or used for internal teaching and non-commercial academic research purposes only. If you are not a member of an academic research institution you must obtain a commercial license; please send requests via email to (suchitj@gene.com).

References

1. Rizvi, N. A. et al. Cancer immunology. mutational landscape determines sensitivity to PD-1 blockade in non-small cell lung cancer. *Science* **348**, 124–128 (2015).
2. van Rooij, N. et al. Tumor exome analysis reveals neoantigen-specific T-cell reactivity in an ipilimumab-responsive melanoma. *J. Clin. Oncol.* **31**, e439–e442 (2013).
3. Yadav, M. et al. Predicting immunogenic tumour mutations by combining mass spectrometry and exome sequencing. *Nature* **515**, 572–576 (2014).
4. Hugo, W. et al. Genomic and transcriptomic features of response to Anti-PD-1 therapy in metastatic melanoma. *Cell* **168**, 542 (2017).
5. Gubin, M. M. et al. Checkpoint blockade cancer immunotherapy targets tumour-specific mutant antigens. *Nature* **515**, 577–581 (2014).
6. Sahin, U. et al. Personalized RNA mutanome vaccines mobilize poly-specific therapeutic immunity against cancer. *Nature* **547**, 222–226 (2017).
7. Sahin, U. et al. An RNA vaccine drives immunity in checkpoint-inhibitor-treated melanoma. *Nature* **585**, 107–112 (2020).
8. Awad, M. M. et al. Personalized neoantigen vaccine NEO-PV-01 with chemotherapy and anti-PD-1 as first-line treatment for non-squamous non-small cell lung cancer. *Cancer Cell* https://doi.org/10.1016/j.ccell.2022.08.003 (2022).
9. Palmer, C. D. et al. Individualized, heterologous chimpanzee adenovirus and self-amplifying mRNA neoantigen vaccine for advanced metastatic solid tumors: phase 1 trial interim results. *Nat. Med.* **28**, 1619–1629 (2022).
10. Jhunjhunwala, S., Hammer, C. & Delamarre, L. Antigen presentation in cancer: insights into tumour immunogenicity and immune evasion. *Nat. Rev. Cancer* **21**, 298–312 (2021).
11. Rock, K. L., Reits, E. & Neefjes, J. Present yourself! by MHC class I and MHC class II molecules. *Trends Immunol.* **37**, 724–737 (2016).
12. Neefjes, J., Jongsma, M. L. M., Paul, P. & Bakke, O. Towards a systems understanding of MHC class I and MHC class II antigen presentation. *Nat. Rev. Immunol.* **11**, 823–836 (2011).
13. Trombetta, E. S. & Mellman, I. Cell biology of antigen processing in vitro and in vivo. *Annu. Rev. Immunol.* **23**, 975–1028 (2005).
14. Abelin, J. G. et al. Mass spectrometry profiling of HLA-associated peptidomes in mono-allelic cells enables more accurate epitope prediction. *Immunity* **46**, 315–326 (2017).
15. Hunt, D. F. et al. Characterization of peptides bound to the class I MHC molecule HLA-A2.1 by mass spectrometry. *Science* **255**, 1261–1263 (1992).
16. Rammensee, H.-G., Friede, T. & Stevanović, S. MHC ligands and peptide motifs: first listing. *Immunogenetics* **41**, 178–228 (1995).
17. Vita, R. et al. The immune epitope database (IEDB) 3.0. *Nucleic Acids Res.* **43**, D405–D412 (2015).
18. Parker, K. C., Bednarek, M. A. & Coligan, J. E. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J. Immunol. Baltim. Md 1950* **152**, 163–175 (1994).
19. Nielsen, M. et al. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Sci.* **12**, 1007–1017 (2003).

20. Nielsen, M. et al. NetMHCpan, a method for quantitative predictions of peptide binding to any HLA-A and -B locus protein of known sequence. *PLoS ONE* **2**, e796 (2007).
21. Karosiene, E., Lundegaard, C., Lund, O. & Nielsen, M. NetMHCcons: a consensus method for the major histocompatibility complex class I predictions. *Immunogenetics* **64**, 177–186 (2012).
22. Vaswani, A. et al. *Attention is All You Need*. in (NeurIPS 2017, 2017).
23. Brown, T. et al. *Language Models are Few-Shot Learners*. in *NeurIPS 2020* (NeurIPS, 2020).
24. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
25. Albert, B. A. et al. Deep neural networks predict class I major histocompatibility complex epitope presentation and transfer learn neoepitope immunogenicity. *Nat. Mach. Intell.* **5**, 861–872 (2023).
26. Wells, D. K. et al. Key parameters of tumor epitope immunogenicity revealed through a consortium approach improve neoantigen prediction. *Cell* **183**, 818–834.e13 (2020).
27. Pyke, R. M. et al. Precision neoantigen discovery using large-scale immunopeptidomes and composite modeling of MHC peptide presentation. *Mol. Cell. Proteomics* **22**, (2023).
28. Sarkizova, S. et al. A large peptidome dataset improves HLA class I epitope prediction across most of the human population. *Nat. Biotechnol.* **38**, 199–209 (2020).
29. Schmidt, J. et al. Prediction of neo-epitope immunogenicity reveals TCR recognition determinants and provides insight into immunoediting. *Cell Rep. Med.* **2**, 100194 (2021).
30. Shao, W. et al. The SysteMHC Atlas project. *Nucleic Acids Res.* **46**, gkx664- (2017).
31. Marcu, A. et al. HLA Ligand Atlas: a benign reference of HLA-presented peptides to improve T-cell-based cancer immunotherapy. *J. Immunother Cancer.* **9**, e002071 (2021).
32. Vita, R. et al. The Immune Epitope Database (IEDB): 2018 update. *Nucleic Acids Res.* **47**, D339–D343 (2019).
33. Rammensee, H.-G., Bachmann, J., Emmerich, N. P. N., Bachor, O. A. & Stevanović, S. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* **50**, 213–219 (1999).
34. O'Donnell, T. J., Rubinsteyn, A. & Laserson, U. MHCflurry 2.0: improved pan-allele prediction of MHC class I-presented peptides by incorporating antigen processing. *Cell Syst.* **11**, 42–48.e7 (2020).
35. Wu, S., Du, Y., Beckford, J. & Alachkar, H. Upregulation of the EMT marker vimentin is associated with poor clinical outcome in acute myeloid leukemia. *J. Transl. Med.* **16**, 170 (2018).
36. Sun, B., Fang, Y., Li, Z., Chen, Z. & Xiang, J. Role of cellular cytoskeleton in epithelial-mesenchymal transition process during cancer progression. *Biomed. Rep.* **3**, 603–610 (2015).
37. Nielsen, M. & Lund, O. NN-align. an artificial neural network-based alignment algorithm for MHC class II peptide binding prediction. *BMC Bioinforma.* **10**, 296 (2009).
38. Bekker, J. & Davis, J. Learning from positive and unlabeled data: a survey. *Mach. Learn.* **109**, 719–760 (2020).
39. Alvarez, B. et al. NNAlign_MA; MHC peptidome deconvolution for accurate MHC binding motif characterization and improved T-cell epitope predictions. *Mol. Cell Proteom.* **18**, 2459–2477 (2019).
40. Bassani-Sternberg, M. & Gfeller, D. Unsupervised HLA peptidome deconvolution improves ligand prediction accuracy and predicts cooperative effects in peptide–HLA interactions. *J. Immunol.* **197**, 2492–2499 (2016).
41. Reynisson, B., Alvarez, B., Paul, S., Peters, B. & Nielsen, M. NetMHCpan-4.1 and NetMHCIIpan-4.0: improved predictions of MHC antigen presentation by concurrent motif deconvolution and integration of MS MHC eluted ligand data. *Nucleic Acids Res* **48**, W449–W454 (2020).
42. Bulik-Sullivan, B. et al. Deep learning using tumor HLA peptide mass spectrometry datasets improves neoantigen identification. *Nat. Biotechnol.* **37**, 55–63 (2019).
43. Rives, A. et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl Acad. Sci.* **118**, e2016239118 (2021).
44. Schuster, H. et al. The immunopeptidomic landscape of ovarian carcinomas. *Proc. Natl Acad. Sci.* **114**, E9942–E9951 (2017).
45. Löffler, M. W. et al. Mapping the HLA ligandome of colorectal cancer reveals an imprint of malignant cell transformation. *Cancer Res.* **78**, canres.1745.2017 (2018).
46. Bassani-Sternberg, M. et al. Deciphering HLA-I motifs across HLA peptidomes improves neo-antigen predictions and identifies allostery regulating HLA specificity. *PLoS Comput. Biol.* **13**, e1005725 (2017).
47. Bassani-Sternberg, M. et al. Direct identification of clinically relevant neoepitopes presented on native human melanoma tissue by mass spectrometry. *Nat. Commun.* **7**, 13404 (2016).
48. Gonzalez-Galarza, F. F. et al. Allele frequency net database (AFND) 2020 update: gold-standard data classification, open access genotype data and new query tools. *Nucleic Acids Res.* **48**, D783–D788 (2020).
49. Shao, X. M. et al. High-Throughput Prediction of MHC Class I and II Neoantigens with MHCnuggets. *Cancer Immunol. Res.* **8**, 396–408 (2020).
50. Garcia Alvarez, H. M., Koşaloğlu-Yalçın, Z., Peters, B. & Nielsen, M. The role of antigen expression in shaping the repertoire of HLA presented ligands. *iScience* **25**, 104975 (2022).
51. Koşaloğlu-Yalçın, Z. et al. Combined assessment of MHC binding and antigen abundance improves T cell epitope predictions. *iScience* **25**, 103850 (2022).
52. Gurung, H. R. et al. Systematic discovery of neoepitope–HLA pairs for neoantigens shared among patients and tumor types. *Nat. Biotechnol.* **42**, 1107–1117 (2024).
53. Ma, B. et al. PEAKS: powerful software for peptide de novo sequencing by tandem mass spectrometry. *Rapid Commun. Mass Spectrom.* **17**, 2337–2342 (2003).
54. Frankish, A. et al. GENCODE 2021. *Nucleic Acids Res.* **49**, D916–D923 (2020).
55. Wu, T. D., Reeder, J., Lawrence, M., Becker, G. & Brauer, M. J. Statistical genomics, methods and protocols. *Methods Mol. Biol. Clifton NJ* **1418**, 283–334 (2016).
56. Pau, G. & Reeder, J. HTSeqGenie: A NGS analysis pipeline. R package version 4.25.1. (2021).
57. Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nat. Methods* **14**, 417–419 (2017).
58. Carbon, S. et al. The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Res.* **49**, D325–D334 (2020).
59. Pearson, H. et al. MHC class I–associated peptides derive from selective regions of the human genome. *J. Clin. Invest.* **126**, 4690–4701 (2016).
60. Cunningham, F. et al. Ensembl 2022. *Nucleic Acids Res.* **50**, D988–D995 (2021).
61. Tareen, A. & Kinney, J. B. Logomaker: beautiful sequence logos in Python. *Bioinformatics* **36**, 2272–2274 (2019).
62. Wagih, O. ggseqlogo: a versatile R package for drawing sequence logos. *Bioinforma. Oxf. Engl.* **33**, 3645–3647 (2017).
63. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. *arXiv* (2019).
64. Howard, J. & Gugger, S. Fastai: A Layered API for Deep Learning. *Information* **11**, 108 (2020).
65. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv* https://doi.org/10.48550/arXiv.1412.6980 (2014).
66. Ramachandran, P., Zoph, B. & Le, Q. V. Searching for activation functions. *arXiv* https://doi.org/10.48550/arXiv.1710.05941 (2017).
67. Henikoff, S. & Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci.* **89**, 10915–10919 (1992).

Acknowledgements

We'd like to thank Aviv Regev for valuable comments and feedback on the manuscript, and Ira Mellman for scientific discussions. All work is funded by Genentech. Figure 1a was created with BioRender.com, released under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International license.

Author contributions

W.J.T., N.L., and S.J. wrote the manuscript with extensive inputs from other authors. S.J. conceptualized the study, designed and directed the data acquisition, data curation, analysis, model development and validation. K.L. guided machine learning method design and analysis. N.L. collected and curated data sets used in the study, developed training and test data, and conducted analyses and model validations. W.J.T and Q.B. developed machine learning models and conducted the analyses and validations. L.D. provided extensive guidance during study design and data acquisition. R.B. provided guidance during model development and validation. J.C. helped in the database design. A.H., E.F., Y.A. and B.H. designed reagents, engineered HMy2.CIR monoallelic cell lines, and conducted validation assays. Q.T.P., C.M.R. and J.R.L. conducted and directed further cell culture of the monoallelic lines, pMHC immunoprecipitation, and mass spectrometry to identify ligands. C.B. provided MHC construct sequences for the study. A-H.C and A-J.T helped with engineered monoallelic cell line assessments and HLA allele prioritization. A-H.C also guided and managed the design of experiments.

Competing interests

The authors of this study are or were employees of Genentech Inc. at the time this work was done. The architecture of the deep learning method described here is related to a US patent filing (US-20220122690-A1) with K.L., N.L., W.J.T, Q.B., L.D., and S.J. as inventors.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-024-54887-7.

Correspondence and requests for materials should be addressed to Kai Liu or Suchit Jhunjhunwala.

Peer review information *Nature Communications* thanks Sandeep Dhanda, Phillip Stafford and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.