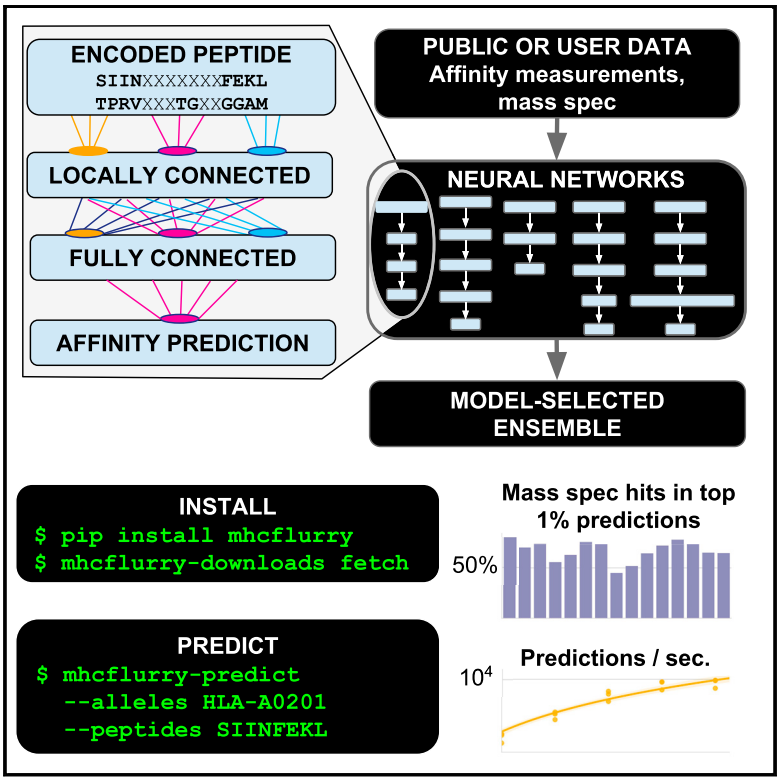
Tool

Cell Systems

MHCflurry: Open-Source Class I MHC Binding Affinity Prediction

Graphical Abstract



Highlights

- Open-source software package for peptide/MHC class I binding prediction
- Easily installed Python package with command line and library interfaces
- Trained on affinity measurements and MHC ligands identified by mass spectrometry

Authors

Timothy J. O'Donnell, Alex Rubinsteyn, Maria Bonsack, Angelika B. Riemer, Uri Laserson, Jeff Hammerbacher

Correspondence

timothy.odonnell@icahn.mssm.edu

In Brief

Accurate prediction servers for MHC I ligands have been in wide use for some time, but these tools are typically closed source, may be trained only by their developers, and can be challenging to integrate into high-throughput workflows required for tumor neoantigen discovery. We introduce a prediction package that exposes a programmatic interface, may be modified and re-retrained, and is much faster than existing tools.

O'Donnell et al., 2018, Cell Systems 7, 129–132
July 25, 2018
https://doi.org/10.1016/j.cels.2018.05.014

CellPress

Cell Systems
Tool

MHCflurry: Open-Source Class I MHC Binding Affinity Prediction

Timothy J. O'Donnell,[1,5,*] Alex Rubinsteyn,[1] Maria Bonsack,[2,3] Angelika B. Riemer,[2,3] Uri Laserson,[1] and Jeff Hammerbacher[4]

[1]Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, NY 10029, USA
[2]Immunotherapy & Immunoprevention, German Cancer Research Center (DKFZ), Heidelberg, Germany
[3]Molecular Vaccine Design, German Center for Infection Research (DZIF), Partner Site Heidelberg, Heidelberg, Germany
[4]Department of Microbiology and Immunology, Medical University of South Carolina, Charleston, SC 29425, USA
[5]Lead Contact
*Correspondence: timothy.odonnell@icahn.mssm.edu
https://doi.org/10.1016/j.cels.2018.05.014

SUMMARY

Predicting the binding affinity of major histocompatibility complex I (MHC I) proteins and their peptide ligands is important for vaccine design. We introduce an open-source package for MHC I binding prediction, MHCflurry. The software implements allele-specific neural networks that use a novel architecture and peptide encoding scheme. When trained on affinity measurements, MHCflurry outperformed the standard predictors *NetMHC 4.0* and *NetMHCpan 3.0* overall and particularly on non-9-mer peptides in a benchmark of ligands identified by mass spectrometry. The released predictor, *MHCflurry 1.2.0*, uses mass spectrometry datasets for model selection and showed competitive accuracy with standard tools, including the recently released *NetMHCpan 4.0*, on a small benchmark of affinity measurements. MHCflurry's prediction speed exceeded 7,000 predictions per second, 396 times faster than *NetMHCpan 4.0*. MHCflurry is freely available to use, retrain, or extend, includes Python library and command line interfaces, may be installed using package managers, and applies software development best practices.

INTRODUCTION

Adaptive immunity depends on T cell recognition of peptides bound to major histocompatibility complex I (MHC I) proteins on cell surfaces. There are thousands of MHC I alleles in the human population, each with specificity for binding a distinct set of peptides, which, when displayed by MHC, can be the target of an immune response. Computational prediction of the binding affinity between a specified peptide and MHC allele has found wide application in infectious diseases, autoimmunity, vaccine design, and cancer immunotherapy (Lundegaard et al., 2007; Ott et al., 2017).

The NetMHC and NetMHCpan tools are considered the state-of-the-art predictive models for this task (Trolle et al., 2015). NetMHC uses an "allele-specific" approach, whereby separate predictors are trained for each MHC allele; the input to the model is the peptide of interest (Andreatta and Nielsen, 2015). NetMHCpan uses a "pan-allele" approach, whereby a single model takes as input both the peptide and a representation of the MHC allele (Nielsen and Andreatta, 2016). Both *NetMHC 4.0* and *NetMHCpan 3.0* are ensembles of shallow neural networks trained on affinity measurements deposited in the immune epitope database (IEDB) (Vita et al., 2015). The recently released *NetMHCpan 4.0* additionally includes peptides eluted from MHC and identified by mass spectrometry (MS) in its training set, generating separate predictions for binding affinity and likelihood of MS identification using two-output neural networks (Jurtz et al., 2017).

We describe and benchmark a package of allele-specific class I MHC binding predictors, *MHCflurry 1.2.0*. MHCflurry predictors show competitive accuracy with the NetMHC tools and a significant speed improvement while addressing a number of limitations. In particular, MHCflurry is open source, publishes the data and workflow used to train models, exposes library and command line interfaces, may be installed using the Python package manager, and applies software development best practices, including unit testing and code documentation.

Implementation

Each supported MHC allele is associated with an ensemble of 8–16 neural networks trained on affinity measurements from IEDB and other sources. In the default MHCflurry predictor (*MHCflurry 1.2.0*), MS data and held-out affinity measurements are used to select 8–16 models for each allele. The software also includes two experimental predictors: *MHCflurry* (*no MS*), which does not use MS datasets, and *MHCflurry* (*train-MS*), which uses MS datasets for both training and model selection (Figure 1A).

MHCflurry supports peptides of length 8–15 using a fixed-length encoding designed to preserve the positionality of the residues that make the most important stabilizing contacts with the MHC. These "anchor positions" occur toward the beginning or end of the peptide for most alleles. Peptides are represented as length-15 sequences, in which missing residues are filled with an *X* character, effectively a 21st amino acid (Figure 1B). The first four and last four residues in the peptide map to the first four and last four positions in the representation. The middle

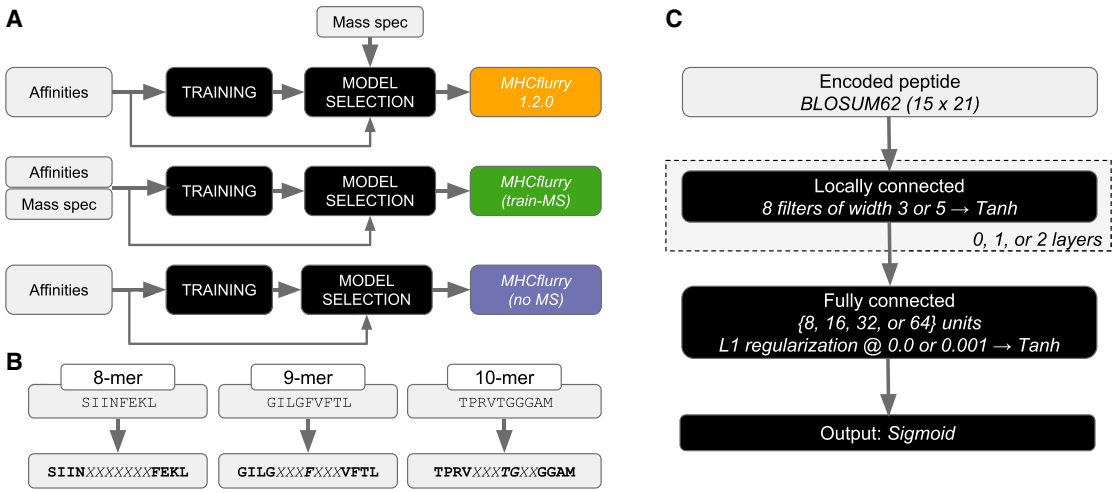


Figure 1. MHCflurry Variants, Peptide Representation, and Neural Network Architectures

(A) Training and model selection schemes for the three MHCflurry variants.

(B) Encodings for three example peptides.

(C) Neural network architectures. Layers with trained weights are shown in black.

seven residues are filled as needed: an 8-mer leaves all middle positions as an *X*, whereas a 15-mer fills all positions. In this way, the positions most likely to contain anchor residues are consistently mapped to the same positions in the representation.

The MHCflurry models are feedforward neural networks composed of the following layers: the length-15 peptide representation with each residue encoded by its vector in the BLOSUM62 substitution matrix (Henikoff and Henikoff, 1992), zero, one, or two locally connected layers, a fully connected layer, and a sigmoidal output (Figure 1C). Locally connected layers are one-dimensional convolutional layers without weight sharing; each neuron receives a neighborhood of adjacent points.

For alleles with fewer than 1,000 training measurements, a pretraining step is used to set the initial model weights. In this step, the training data are augmented with measurements from similar alleles to make a set of at least 1,000 measurements, where similarity is in terms of the BLOSUM62 similarity score of the binding-core residues of the allele. The model is first trained using the augmented training set, then retrained on the non-augmented set starting from the weights learned in the first step.

To train and select MHCflurry models, we assembled a dataset of 230,735 affinity measurements across 130 alleles from IEDB and a published benchmark (Kim et al., 2014). Ten percent of this data plus 226,684 MS-identified ligands deposited in IEDB or the SysteMHC Atlas (Shao et al., 2018), or published elsewhere (Abelin et al., 2017), were used to perform model selection. Of 130 alleles for which preliminary full ensembles were trained, 112 showed sufficient performance on the model selection dataset to include in the final predictor. For these 112 alleles, the model selection dataset was used to select 8–16 of the 320 total models trained per allele using a forward stepwise selection procedure (Caruana et al., 2004).

MHCflurry predicts quantitative binding affinities, but one-fourth of the entries (57,828 of 230,735) in the affinity dataset are qualitative, represented as *positive*, *positive-high*, *positive-intermediate*, *positive-low*, or *negative*. To use these measurements, the MHCflurry models are trained using a modification to the mean square error (MSE) loss function, whereby measurements may be associated with an inequality, (>) or (<), and contribute to the loss only when the inequality is violated. For example, we assigned measurements represented as *positive-high* the value "<100 nM," as such peptides are likely to have binding affinities tighter than (i.e., less than) 100 nM. During training, these peptides contribute to the loss only when their predictions are greater than 100 nM. For the *MHCflurry* (*train-MS*) predictor, this approach is used to include MS-identified ligands, which are assigned a "<500 nM" value.

RESULTS

As the predictors considered have similar accuracy, power to identify differences requires a large held-out validation dataset. However, nearly all published measurements are potentially included in the training data for the NetMHC tools. We therefore adopted a strategy based on two benchmarks, whereby first a large recently published MS dataset is used to benchmark the *MHCflurry* (*no MS*) predictor against versions of the NetMHC tools trained without MS, *NetMHC 4.0*, and *NetMHCpan 3.0*. In a second benchmark, we apply a smaller dataset of unpublished affinity measurements to assess accuracy across all tools, including the default *MHCflurry 1.2.0* predictor and *NetMHCpan 4.0*, which include MS datasets in their training pipelines. In the first benchmark, we find that MHCflurry outperforms the other predictors due to improved accuracy on non-9-mer peptides. In the smaller benchmark, *MHCflurry 1.2.0* narrowly outperformed *NetMHC 4.0* and *MHCflurry* (*no MS*), but performed equivalently within statistical uncertainty to *MHCflurry* (*train-MS*), *NetMHCpan 3.0*, and *NetMHCpan 4.0*.

The MS benchmark consists of a published set of 23,651 MHC ligands (Abelin et al., 2017). We scored predictors by their positive predictive value (PPV) at differentiating MS-identified peptides from decoys sampled from the same transcripts (STAR Methods).

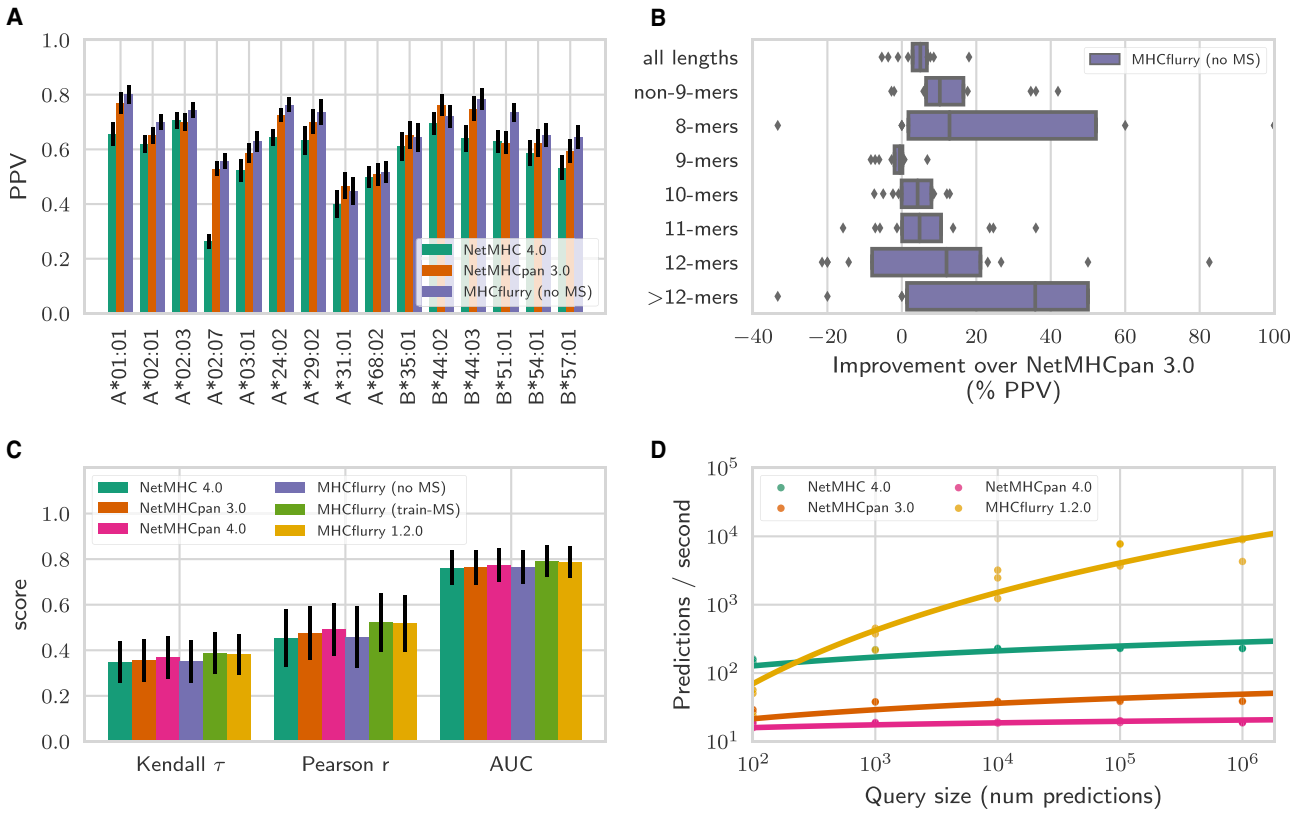


Figure 2. Benchmark Results

(A) Positive predictive value (PPV) of *NetMHC 4.0*, *NetMHCpan 3.0*, and the *MHCflurry* (*no MS*) predictors on the mass spectrometry (MS) benchmark. The *NetMHCpan 4.0*, *MHCflurry* (*train-MS*), and *MHCflurry 1.2.0* predictors are excluded from this evaluation because their training or model selection datasets include MS data.

(B) *MHCflurry* (*no MS*) accuracy difference with respect to *NetMHCpan 3.0* for each peptide length across alleles in the MS benchmark. The median line is indicated, boxes show quartiles, and points indicate alleles outside the interquartile region. The >12-mers category includes 13-, 14-, and 15-mers.

(C) Kendall rank correlation coefficient, Pearson correlation over log affinities, and area under the receiver-operating characteristic curve (AUC) on the HPV dataset.

(D) Prediction speed on subsets of the MS dataset consisting of a single allele and a varying number of peptides.

Error bars in (A) and (C) indicate bootstrap 95% confidence intervals. See also Tables S1 and S2.

MHCflurry (*no MS*) showed improved accuracy over *NetMHC 4.0* and *NetMHCpan 3.0*, outperforming these predictors on 15/15 and 12/15 alleles, respectively (binomial test $p < 0.02$ for both tests; Figure 2A and Table S1). Across alleles, MHCflurry showed a median 16.1% (range 3.8–110.5) higher PPV than *NetMHC 4.0* and 5.0% (−5.4 to +18.0) higher PPV than *NetMHCpan 3.0*.

The accuracy advantage of MHCflurry over the NetMHC tools was due to better performance on non-9-mer peptides, which offset nominally lower accuracy on 9-mers relative to *NetMHCpan 3.0* (Figure 2B). On non-9-mers (i.e., peptides of lengths 8 and 10–15), MHCflurry outperformed *NetMHC 4.0* on 15/15 alleles (median PPV improvement 34.1%; range 7.1–224.4; $p < 0.0001$) and *NetMHCpan 3.0* on 13/15 alleles (median PPV improvement 10.2%; range −2.8 to +41.9; $p = 0.004$). On 9-mer peptides, no tool showed significant performance improvement over the others: *MHCflurry* (*no MS*) outperformed *NetMHC 4.0* on 9/15 alleles ($p = 0.3$), with a median difference in PPV of 0.5% (range −6.4 to +9.0), but underperformed *NetMHCpan 3.0* on 10/15 alleles ($p = 0.15$) with a median difference in PPV of −0.5% (range −8.3 to +6.9).

The second benchmark uses an unpublished set of affinity measurements for 475 human papillomavirus 16 (HPV16)-derived peptides in cell-based competitive binding assays across seven MHC alleles. Several accuracy metrics were computed for each predictor (Figure 2C and STAR Methods). While the small size of the dataset led to high uncertainties in absolute scores, an analysis of the differences in score relative to *MHCflurry 1.2.0* within bootstrap resamples indicated that *MHCflurry 1.2.0* outperformed *NetMHC 4.0* and *MHCflurry* (*no MS*) on some metrics (Table S2). However, no differences were detected between *MHCflurry 1.2.0* and the *NetMHCpan 4.0*, *NetMHCpan 3.0*, or *MHCflurry* (*train-MS*) predictors.

When running a large numbers of predictions, MHCflurry was substantially faster than the other predictors (Figure 2D). Using only the CPU, *MHCflurry 1.2.0* approached 7,500 predictions per second, 396 times faster than *NetMHCpan 4.0* and 12 times faster than *NetMHC 4.0*. Use of a graphics processing unit did not substantially improve MHCflurry performance.

DISCUSSION

MHCflurry is an open-source package for MHC I affinity prediction with a fast and documented implementation. On the MS benchmark, *MHCflurry* (*no MS*) outperformed the *NetMHC 4.0* and *NetMHCpan 3.0* tools overall and particularly on non-9-mer peptides. While part of this advantage may be due to improvements such as explicit support for variable length peptides, differences in training datasets between the tools, which are difficult to assess as those for the NetMHC tools are not released, likely also contribute.

As MHCflurry is an allele-specific predictor, only a fixed set of alleles are supported. Pan-allele predictors such as NetMHCpan remain the best option for alleles with few data. However, the data required to fit a MHCflurry model can be modest. For example, the A*02:07 allele has 126 measurements in the affinity measurement set, the fewest of any allele tested in the MS benchmark. As expected, *NetMHC 4.0* performs poorly, with a PPV of 0.26 on this benchmark. The *MHCflurry* (*no MS*) predictor, however, performs respectably (PPV = 0.56), in fact narrowly outperforming *NetMHCpan 3.0* (PPV = 0.53). This is largely due to the pre-training step used in MHCflurry, which enables models to borrow information from similar alleles.

The standard *MHCflurry 1.2.0* predictor is trained on affinity measurements and uses MS ligands only for model selection. This is a conservative choice that minimizes the potential impact of any biases associated with ligands identified by MS, such as depletion of cysteines (Abelin et al., 2017). However, the *MHCflurry* (*train-MS*) predictor, which includes MS in its training set, showed good performance in the HPV benchmark and may eventually become the default MHCflurry predictor.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- CONTACT FOR REAGENT AND RESOURCE SHARING
- METHOD DETAILS
  - Model Inputs and Outputs
  - Training
  - Model Selection
  - Construction of Training and Validation Datasets
- QUANTIFICATION AND STATISTICAL ANALYSIS
- DATA AND SOFTWARE AVAILABILITY

SUPPLEMENTAL INFORMATION

Supplemental Information includes two tables and can be found with this article online at https://doi.org/10.1016/j.cels.2018.05.014.

ACKNOWLEDGMENTS

We thank Mike Rooney for helpful discussions. Generation of the HPV dataset was supported by intramural funding of the German Cancer Research Center (DKFZ), grant number TTU 07.706 by the German Center for Infection Research (DZIF) to A.B.R., and a PhD scholarship to M.B. by the Helmholtz International Graduate School of the DKFZ. This work was supported by the Parker Institute for Cancer Immunotherapy.

AUTHOR CONTRIBUTIONS

T.J.O. and A.R. developed MHCflurry. T.J.O. benchmarked the software and wrote the paper. M.B. and A.B.R. performed the HPV peptide binding experiments and advised on benchmarking approaches. T.J.O., U.L., and J.H. supervised the project. All authors critically reviewed the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 19, 2017
Revised: March 30, 2018
Accepted: May 21, 2018
Published: June 27, 2018

REFERENCES

Abelin, J.G., Keskin, D.B., Sarkizova, S., Hartigan, C.R., Zhang, W., Sidney, J., Stevens, J., Lane, W., Zhang, G.L., Eisenhaure, T.M., et al. (2017). Mass spectrometry profiling of HLA-associated peptidomes in mono-allelic cells enables more accurate epitope prediction. Immunity *46*, 315–326.

Andreatta, M., and Nielsen, M. (2015). Gapped sequence alignment using artificial neural networks: application to the MHC class I system. Bioinformatics *32*, 511–517.

Caruana, R., Niculescu-Mizil, A., Crew, G., and Ksikes, A. (2004). Ensemble selection from libraries of models. In Proceedings of the Twenty-first International Conference on Machine Learning, C.E. Brodley, ed. (ACM), p. 18.

Henikoff, S., and Henikoff, J.G. (1992). Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. USA *89*, 10915–10919.

Jurtz, V., Paul, S., Andreatta, M., Marcatili, P., Peters, B., and Nielsen, M. (2017). NetMHCpan-4.0: improved peptide-MHC class I interaction predictions integrating eluted ligand and peptide binding affinity data. J. Immunol. *199*, 3360–3368.

Kessler, J.H., Mommaas, B., Mutis, T., Huijbers, I., Vissers, D., Benckhuijsen, W.E., Schreuder, G.M.T., Offringa, R., Goulmy, E., Melief, C.J.M., et al. (2003). Competition-based cellular peptide binding assays for 13 prevalent HLA class I alleles using fluorescein-labeled synthetic peptides. Hum. Immunol. *64*, 245–255.

Kessler, J.H., Benckhuijsen, W.E., Mutis, T., Melief, C.J.M., van der Burg, S.H., and Drijfhout, J.W. (2004). Competition-based cellular peptide binding assay for HLA class I. Curr. Protoc. Immunol. https://doi.org/10.1002/0471142735.im1812s61.

Kim, Y., Sidney, J., Buus, S., Sette, A., Nielsen, M., and Peters, B. (2014). Dataset size and composition impact the reliability of performance benchmarks for peptide-MHC binding predictions. BMC Bioinformatics *15*, 241.

Lundegaard, C., Lund, O., Keşmir, C., Brunak, S., and Nielsen, M. (2007). Modeling the adaptive immune system: predictions and simulations. Bioinformatics *23*, 3265–3275.

Nielsen, M., and Andreatta, M. (2016). NetMHCpan-3.0; improved prediction of binding to MHC class I molecules integrating information from multiple receptor and peptide length datasets. Genome Med. *8*, 33.

Ott, P.A., Hu, Z., Keskin, D.B., Shukla, S.A., Sun, J., Bozym, D.J., Zhang, W., Luoma, A., Giobbie-Hurder, A., Peter, L., et al. (2017). An immunogenic personal neoantigen vaccine for patients with melanoma. Nature *547*, 217–221.

Shao, W., Pedrioli, P.G.A., Wolski, W., Scurtescu, C., Schmid, E., Vizcaíno, J.A., Courcelles, M., Schuster, H., Kowalewski, D., Marino, F., et al. (2018). The SysteMHC Atlas project. Nucleic Acids Res. *46*, D1237–D1247.

Trolle, T., Metushi, I.G., Greenbaum, J.A., Kim, Y., Sidney, J., Lund, O., Sette, A., Peters, B., and Nielsen, M. (2015). Automated benchmarking of peptide-MHC class I binding predictions. Bioinformatics *31*, 2174–2181.

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

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Chemicals, Peptides, and Recombinant Proteins		
8-11-mer peptides derived from HPV type 16 Protein E6	DKFZ Genomics and Proteomics Core Facility – Peptide Synthesis	UniProtKB - P03126
8-11-mer peptides derived from HPV type 16 Protein E7	DKFZ Genomics and Proteomics Core Facility – Peptide Synthesis	UniProtKB - P03129
β2-microglobulin	MP Biomedicals EMEA, Illkirch, France	Cat#153903.5 MDL#MFCD00130615
Deposited Data		
IEDB affinity data	Vita et al., 2015	http://www.iedb.org/doc/mhc_ligand_full.zip Downloaded on Dec. 1, 2017
BD2013	Kim et al., 2014 http://tools.iedb.org/main/datasets/	http://tools.iedb.org/static/main/benchmark_mhci_reliability.tar.gz
Peptides eluted from MHC I and identified by mass spec	Abelin et al., 2017	Table S1
MHCflurry 1.2.0 models	This paper; and Mendeley Data	https://doi.org/10.17632/8pz43nvvxh.1
MHCflurry (no MS) models	This paper; and Mendeley Data	https://doi.org/10.17632/8pz43nvvxh.1
MHCflurry (train-MS) models	This paper; and Mendeley Data	https://doi.org/10.17632/8pz43nvvxh.1
Curated training and model selection dataset	This paper; and Mendeley Data	https://doi.org/10.17632/8pz43nvvxh.1
MS benchmark dataset	This paper; and Mendeley Data	https://doi.org/10.17632/8pz43nvvxh.1
Experimental Models: Cell Lines		
1341-8346	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW01060
BSM	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09032
E481324	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09011
EA	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09081
FH8	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09382
LKT3	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09107
WT100BIS	International Histocompatibility Working Group Cell Bank, Seattle, WA, USA	IHW09006
Software and Algorithms		
MHCflurry 1.2.0	This paper	https://github.com/openvax/mhcflurry
NetMHC 4.0	Andreatta and Nielsen, 2015	http://www.cbs.dtu.dk/cgi-bin/sw_request?netMHC
NetMHCpan 3.0	Nielsen and Andreatta, 2016	http://www.cbs.dtu.dk/cgi-bin/sw_request?netMHCpan+3.0
NetMHCpan 4.0	Jurtz et al., 2017	http://www.cbs.dtu.dk/cgi-bin/nph-sw_request?netMHCpan
Keras 2.1.2	Github	https://github.com/keras-team/keras
Other		
HPV dataset	This paper	Available upon request to not-for-profit enterprises for research purposes only

CONTACT FOR REAGENT AND RESOURCE SHARING

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Timothy O'Donnell (timothy.odonnell@icahn.mssm.edu).

METHOD DETAILS

MHCflurry is implemented in Python using the Keras neural network library (https://github.com/keras-team/keras). In our experiments we used the tensorflow backend (https://www.tensorflow.org).

Model Inputs and Outputs

Each allele is associated with an ensemble of 8-16 neural networks selected from 320 models. The predicted nanomolar affinity is taken to be the geometric mean of the individual model outputs. The variance of the predictions across the ensemble gives an indication of uncertainty and is made available to users, as is the quantile of each prediction among a large set of random peptides (100,000 for each length 8-15) pre-computed for each allele.

As in the NetMHC tools, MHCflurry internally transforms binding affinities to values between 0.0 and 1.0, where 0.0 is a non-binder and 1.0 is a strong binder. The neural networks are trained using the transformed values and the inverse transform is used to return prediction results as nanomolar affinities. The transform is given by $1 - log_{50000}(x)$ where x is the nanomolar affinity. Affinities are capped at 50,000 nM.

The input to each MHCflurry model is a peptide encoded as a 15 x 21 matrix. Rows give the BLOSUM62 encoding of each residue in the peptide, after transforming it to a length-15 sequence as described. As BLOSUM62 is a substitution matrix, amino acids are thus represented by their similarity to the other amino acids (Henikoff and Henikoff, 1992). We define the special "no residue" *X* character used in the 15-mer peptide representation to have 0.0 similarity to all other amino acids and 1.0 similarity to itself. No representation of the MHC allele, such as its amino acid sequence, is provided to the network.

Training

Models are drawn from 40 architectures trained on either (1) all affinity data for an allele or (2) only quantitative affinity data. For each of these 80 possibilities, four replicates are trained, for a total of 320 models per allele.

Training is attempted for all alleles with at least 25 affinity measurements. Ten percent of the training data is set aside for model selection. Each neural network is trained on a different 90% sample of the remaining data, with the other 10% used as a test set for early stopping. Training proceeds with the RMSprop optimizer using a minibatch size of 128 until the accuracy on the test set has not improved for 20 epochs. At each epoch, 25 synthetic negative peptides for each length 8–15 are randomly generated. These random negative peptides are sampled so as to have the same amino acid distribution as the training peptides and are assigned affinities >20,000 nM. For the *MHCflurry* (*train-MS*) variant, the number of random peptides for each length is $0.2n + 25$ where n is the number of training peptides.

A modified mean squared error (MSE) loss function that supports data with inequalities is used for both the training loss and test set accuracy metric. For this loss function, measurements are associated with an inequality: (<), (>), or (=). The loss L is defined as:

$$L(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{n}\sum_{i}^{n} I(\hat{y}_i, y_i)$$

$$I(\hat{y}_i, y_i) = \begin{cases} (max(\hat{y}_i - y_i, 0))^2 & \text{if inequality for measurement } i \text{ is } (<) \\ (max(y_i - \hat{y}_i, 0))^2 & \text{if inequality for measurement } i \text{ is } (>) \\ (\hat{y}_i - y_i)^2 & \text{if inequality for measurement } i \text{ is } (=) \end{cases}$$

where n is the total number of measurements, and $\hat{y}_i$ and y_i are the predicted and measured values for measurement i, respectively. Quantitative affinity data is associated with an inequality of (=). For qualitative affinity data, we assigned the following inequalities and measurement values: *positive-high*, < 100 nM; *positive*, < 500 nM, *positive-intermediate*, < 1,000 nM; *positive-low*, < 5,000 nM; *negative*, > 5,000 nM. In the *MHCflurry* (*train-MS*) variant, MS-identified ligands are assigned the value "< 500 nM."

Model Selection

Model selection for *MHCflurry 1.2.0* uses the 10% of affinity measurements held out during training augmented by ligands identified in MS elution experiments. Ensembles are evaluated using the variant of mean squared error previously described for affinity data and positive predictive value (PPV) on MS data. The decoy set used for the MS evaluation consists of all other peptides identified by MS for an allele other than the one in question. The final score is the average of the MSE and PPV scores weighted by the number of observations contributing to each.

As the first step in model selection, the full ensembles (320 models per allele) are evaluated to identify alleles for which training failed to give an acceptable predictor, which is generally due to insufficient data. This is done using a permutation test, requiring

that the score of the actual predictions of the full ensemble fall above the 95th percentile among the scores obtained by randomly permuting the measurement labels. This can be interpreted as rejecting the null hypothesis that the ensemble is a random predictor. Alleles for which we cannot reject this null hypothesis are excluded from model selection and are unsupported by the standard MHCflurry predictor. From the 130 alleles for which training was attempted, this filter reduced the supported alleles to 112.

Ensembles are selected using a forward stepwise selection procedure (Caruana et al., 2004). Starting with an empty ensemble, models are added to maximize the score of the ensemble at each step, stopping when no model's addition improves the score. We require that at least 8 and no more than 16 models are selected per allele. To help reduce the noise associated with model selection on alleles with limited data, we additionally include in the combined score the consistency with the predictions of the full ensemble of 320 models (in terms of Kendall τ) on a large number of random peptides, weighting it be equivalent to 10 affinity measurements or MS hits.

Construction of Training and Validation Datasets

Affinity Measurements Used for Training and Model Selection

The affinity measurement dataset used for training and model selection was assembled from a snapshot of the Immune Epitope Database (IEDB) MHC ligands downloaded on Dec. 1, 2017 augmented with the BD2013 dataset (Kim et al., 2014). IEDB entries with non-class I, non-specific, mutant, or unparseable allele names were dropped, as were those with peptides identified by MS or containing post-translational modifications or noncanonical amino acids. This yielded an IEDB dataset of 143,898 quantitative and 43,978 qualitative affinity measurements. Of 179,692 measurements in the BD2013 dataset (Kim et al., 2014), 57,506 were not also present in the IEDB dataset. After selecting peptides of length 8-15 and dropping alleles with fewer than 25 measurements, the combined dataset consists of 230,735 measurements across 130 alleles.

MS Model Selection Dataset

The MS model selection dataset consists of 226,684 ligands formed by combining 186,415 ligands from IEDB with 39,741 additional ligands from SysteMHC Atlas (Shao et al., 2018) and 530 additional ligands from ref. (Abelin et al., 2017). The unprocessed SysteMHC and Abelin et al. datasets are much larger but most entries are already present in IEDB, are duplicates, or report on alleles for which training was not attempted. The ligands from SysteMHC Atlas were first filtered to remove entries with low confidence (*prob* < *0.99*). Of the 112 alleles supported by the predictor, 57 had at least 100 MS ligands available for model selection (Table S1).

MS Benchmark Dataset

The MS benchmark was derived from 23,651 sequences of MHC-displayed ligands eluted from a B cell line expressing a single MHC I allele (Abelin et al., 2017). We excluded one allele (HLA-A*02:04) not supported by MHCflurry or NetMHC due to insufficient representation in the training dataset (fewer than 25 measurements) and discarded peptides with post-translational modifications or lengths outside the supported range (8-15 residues). We sampled unobserved sequences (decoys) from the protein-coding transcripts that contained the identified peptides (hits) based on protein sequences in the UCSC hg19 proteome and transcript quantification from RNA sequencing of the relevant cell line (B721.221) downloaded from the Gene Expression Omnibus (accession GSE93315). For an allele with *n* hits, we sampled 100*n* decoys, weighting transcripts by the number of hits and sampling an equal number of decoys of each length 8-15. After removing any entries also present in the training dataset, this yielded a benchmark of 23,651 hits and 2,377,042 decoys.

HPV Affinity Measurement Benchmark Dataset

The HPV benchmark dataset consists of affinity measurements for 475 peptides of length 8-11 across seven MHC alleles (HLA-A*01:01, HLA-A*02:01, HLA-A*03:01, HLA-A*11:01, HLA-A*24:02, HLA-B*07:02, HLA-B*15:01).

The binding affinity of HPV16 E6 and E7 derived peptides to selected MHC class I molecules was tested in competition-based binding assays as described in (Kessler et al., 2003, 2004). Briefly, test peptides in 1:2 serial dilutions (final concentrations from 100 – 0.78 μM) compete with 150 mM fluorescein-labeled reference peptide with a known high affinity for binding to the MHC class I molecule of interest on B-LCL cells (cell lines used: 1341-8346, BSM, E481324, EA, FH8, LKT3, WT100BIS), which were previously stripped from natural bound peptides and β2-microglobulin using ice cold citric acid buffer. After stripping, the cells were washed with culture medium and dissolved in culture medium containing 2 μg/mL β2-microglobulin (MP Biomedicals) to reconstitute the MHC class I complex. B-LCL cells were diluted to $6x10^4$ cells/100 μl per test peptide concentration and pipetted to a well-plate containing the mixes of test and reference peptide. After 24h incubation at 4°C the cells were washed, fixed in 1% PFA and suspended in 0.5% BSA in 1x PBS. The mean fluorescence intensity *Fmix* at each test peptide concentration was measured by flow cytometry (Accuri C6, BD Biosciences). The binding of each test peptide was calculated as the percent inhibition of reference peptide binding relative to the minimal response (without reference; *Fmin*) and the maximal response (reference only; *Fmax*) as:

$$Inhibition\ (\%) = \left(1 - \frac{Fmix - Fmin}{Fmax - Fmin}\right) * 100$$

The binding affinity of the test peptide was determined by non-linear regression analysis as the concentration that inhibits 50% binding of the fluorescein-labeled reference peptide (IC50).

Peptides with an experimental IC50 below 5 μM were defined as strong binders, 5-15 μM as intermediate binders, 15-100 μM weak binders, and peptides above 100 μM were defined as non-binders, as outlined in (Kessler et al., 2003, 2004). These rather high

nM-values are explained by the fact that the experimental assay uses very high affinity reference peptides, thus high concentrations of test peptides are needed to reach the IC50. For confirmation and statistical significance the assay was performed at least three times for binders and twice for non-binders.

Speed Benchmarking

Experiments were performed on a machine with twelve Intel Core(TM) i7-5930K CPUs at 3.50GHz, four NVIDIA GeForce GTX TITAN X GPUs, and 64GB memory using the MHCtools Python interface to the MHCflurry and NetMHC tools (https://github.com/openvax/mhctools) with parallelization and GPUs disabled. We measured the time to generate various numbers of predictions (10^2, 10^3, 10^4, 10^5, and 10^6) for a single allele using peptides sampled from the MS benchmark. We repeated the experiment three times using different alleles (HLA-A*02:01, HLA-A*02:07, HLA-A*01:01). Rates and speedups reported in the main text are averages for the three alleles at the maximum number of peptides (10^6).

Training the *MHCflurry 1.2.0* full ensembles (320 models for each of 130 alleles, for 41,600 models total) took 1,049 minutes using all GPUs and CPUs on the machine. Model selection took 299 minutes, and computing the histogram of predicted affinities for each allele took 15 minutes.

QUANTIFICATION AND STATISTICAL ANALYSIS

Accuracy on the MS benchmark was assessed in terms of the positive predictive value (PPV) for the predicted affinity to differentiate MS hits from decoys. This was calculated separately for each allele. To compute PPV for an allele with n hits, we ranked the $n + 100n$ hits and decoys from tightest to weakest predicted binding affinity and calculated the fraction of the top n peptides that were hits. Scores relative to NetMHC or NetMHCpan were calculated by dividing the difference in PPV between MHCflurry and the other predictor by the other predictor's PPV and representing the result as a percent. We applied a two-sided binomial test (α=0.05) to determine if one predictor outperformed another on more alleles than expected by chance.

We assessed accuracy on the HPV affinity measurement dataset using three metrics: Kendall rank correlation coefficient (Kendall τ), Pearson correlation taken over the log of the predicted and measured affinities, and the area under the receiver operator characteristic curve (AUC) at differentiating binders from non-binders. As the HPV dataset is small, these metrics were computed across all predictions, not separately for each allele. Kendall τ measures the correlation in rank when peptides are sorted by measured or predicted affinity. Kendall τ and Pearson correlation were calculated using the scipy package (https://www.scipy.org). The AUC estimates the probability that a binding peptide will have a stronger predicted affinity than a non-binding peptide. It was calculated using the scikit-learn package (http://scikit-learn.org). For the purpose of AUC, we defined a peptide to be a binder if it had any detectable binding in our assay (Figure 2C). AUCs calculated using more restrictive IC50 thresholds are indicated in Table S2. Each predictor was compared to *NetMHCpan 4.0* on each metric by computing the difference of the two predictors' scores within bootstrap resamples of the dataset, with a result considered significant if the 95% confidence interval for the difference excludes 0.

DATA AND SOFTWARE AVAILABILITY

MHCflurry is available under the Apache License 2.0. It may be installed from the Python package index. Source code is maintained at https://github.com/openvax/mhcflurry. All data and scripts used to train the models are available in this repository. The trained models as well as the training and MS datasets are also deposited at https://doi.org/10.17632/8pz43nvvxh.1. The HPV dataset includes unpublished affinity measurements (Hoppe, Bonsack et al., manuscript in preparation) and are available upon request to not-for-profit enterprises for research purposes only.