Positional SHAP for Interpretation of Deep Learning Models Trained from Biological Sequences

Quinn Dickinson, Jesse G. Meyer

Department of Biochemistry, Medical College of Wisconsin, Milwaukee, Wisconsin, 53226

Abstract
Machine learning with artificial neural networks, also known as “deep learning”, accurately predicts biological phenomena such as disease diagnosis and protein structure. Despite the ability of deep learning to make accurate biological predictions, a challenge is model interpretation, which is especially challenging for recurrent neural network architectures due to the sequential input data. Here we train multi-output long short-term memory (LSTM) regression models to predict peptide binding affinity to five rhesus macaque major histocompatibility complex (MHC) I alleles. We adapt SHapely Additive exPlanations (SHAP) to generate positional model interpretations of which amino acids are important for peptide binding. These positional SHAP values reproduced known rhesus macaque MHC class I (Mamu-A1*001) peptide binding motifs and provided insights into inter-positional dependencies of peptide-MHC interactions. Positional SHAP should find widespread utility for interpreting a variety of models trained from biological sequences.

Keywords: Neural networks, deep learning, model interpretation, SHAP, peptides, MHC, LSTM

INTRODUCTION

Sequences are ubiquitous throughout the biological world. Nucleic acids and proteins encode information as sequences of monomeric building blocks. Sequence order is extremely important; the primary amino acid sequence of a protein uniquely determines the set of 3D structures formed after folding. Decades of effort by thousands of scientists has focused on measuring protein structures[1–3] and determining intermolecular binding[4]. Significant efforts have been devoted to protein structure prediction[5,6]. Recent advances in deep learning have achieved major milestones in protein structure prediction[7].

Deep learning is a type of machine learning that uses neural networks to learn complex relationships between pairs of input and output data[8]. There are many types of neural network models that differ primarily in how neurons are connected. Each architecture is well suited for different types of input data. For example, convolutional neural networks (CNNs) are effective at using images as inputs[9], and recurrent neural networks (RNNs) are effective at using sequence data as input[10]. RNNs have found extensive application to natural language processing, and by extension as a similar type of data, predictions from biological sequences such as peptides or nucleic acids. A specific type of RNN called a long short-term memory (LSTM) solves the vanishing gradient problem seen with backpropagation of standard RNNs[11], and thus LSTMs are widely used for sequence data.

One goal of building predictive models is to create an understandable and actionable relationship between the input and output data. Although deep learning with LSTM models is effective for making predictions from sequences, a challenge is interpreting how input data leads to specific outputs. For certain models, like CNNs, there are model specific interpretation strategies, such as layer-wise relevance propagation[12]. Additionally, there are several strategies to enable interpretation of any arbitrary model type, such as permutation importance[13], and SHapley Additive exPlanations (SHAP)[14,15]. SHAP is an attractive option because, in addition to it working on any arbitrary model, SHAP can dissect interactions between inputs when they are correlated. However, SHAP does not directly enable sequence-dependent model interpretation, which is needed for understanding how RNN or LSTM models connect biological sequence inputs to specific predictions.

The major histocompatibility complex (MHC) is an array of closely related genes that encode cell surface proteins that form an essential part of the adaptive immune system[16–18]. There are two main classes of MHC complexes, I and II. MHC class I molecules are expressed on the cell surface of all nucleated cells in vertebrates. MHC class I molecules present endogenous peptides that are recognized by the CD8 T cell receptors (TCR) on cytotoxic T cells. If the cytotoxic T cells recognize peptides that are non-self, such as from a viral infection or cancer, T cells will induce apoptosis in the abnormal cell.

Peptides bound by the MHC I complex are primarily generated by the proteasome from intracellular proteins[19]. Not all peptides from proteasome degradation are bound into the MHC class I complex, nor are peptides bound with equal frequency. Peptides suitable for the MHC class I complex are generally between eight and ten amino acids in length, although longer peptides have been reported[20]. The sequence of the peptide determines its binding affinity to each MHC class I complex allele. Given the polymorphism of MHC class I alleles in the human population, the abundance of potential binding peptides, and the low throughput of current binding assays, direct measurement of peptide-MHC binding for all possible peptides is infeasible. Therefore, the prediction of binding affinities through methods such as machine learning or molecular modelling could lead to improved development of vaccines against diseases like cancer[21].

Extensive efforts have focused on prediction of peptide-MHC interactions[22,23]. Both classification and regression models are used to learn which peptides bind an MHC allele (for example, see O'Donnell *et al.*, Zeng and Gifford, and Liu *et al.*)[24–26]. Further work is needed to interpret predictive models to understand general

patterns of amino acids that will bind a given MHC I allele. Here we demonstrate that LSTM models can easily learn to perform regression directly from a peptide sequence to that peptide's affinity to various MHC I alleles (measured by Haj *et al.* 2020)[27]. Our main contribution is a strategy to interpret such models that we term "positional SHAP", which reveals how each amino acid contributes to binding specifically and generally across all peptide predictions. We extend the strategy to track inter-positional dependence of amino acids for binding. This work therefore describes a general and broadly applicable framework for understanding notoriously abstruse deep learning models trained on biological sequences.

METHODS

Data

Data used for training and testing the model was obtained from Haj *et al.* 2020[27], where all possible 8-, 9-, and 10-mer peptides from 82 simian immunodeficiency virus (SIV) and simian-human immunodeficiency virus (SHIV) strains were measured by fluorescent peptide array. The data provided consists of 61,066 entries containing the peptide sequence, the peptide length, and five intensity values from the fluorescence assay for each of the five Mamu alleles tested (A001, A002, A008, B008, and B017). These intensity values are a proxy for binding affinity between a peptide and the MHC I allele. From the methods of Haj *et al.*, each intensity is the base 2 logarithm of the median intensity value of five replicates reported for each peptide as measured by an MS200 Scanner at a resolution of 2µm and a wavelength of 635nm[27]. For training and testing of the model, the dataset was randomly split into three categories (**Supplemental Figure 1**). Because the dataset contains truncated forms of each core peptide sequence as 8-, 9-, and 10-mers, the data splitting grouped each core sequence into unique indexes and split based on those indexes. This core sequence-based splitting ensured that training and testing data would not have truncated versions of the same peptide. The training data had 43,923 entries (71.93% of all data). The validation data to assess overfitting during training had 10,973 entries (17.97% of all data). Test data consisted of 6,170 entries (10.10% of all data) and was used after hyperparameter optimization to assess the overall accuracy of the model.

Model architecture

The Keras (version 2.3.0-tf)[28] interface for Tensorflow (version 2.2.0)[29] was used to build and train the LSTM model (see detailed model architecture in **Supplemental Figure 2**). Peptide sequences were converted to integers ranging from 0 to 21 where each integer corresponds to an amino acid or the special token “END”, which is used to pad peptides with length 8 or 9 to have length 10. The embedding layer takes these ten integer inputs corresponding to each position of the peptide. Each input is transformed by the embedding layer learn a 10x50 dimensional representation that is sent to the first LSTM layer[11]. The LSTM layer outputs a 50x128 dimensional tensor to a dropout layer where 44% of values are randomly set to 0 before passing to a second LSTM layer that outputs a tensor with length 128. A second dropout layer then randomly sets 20% of values to 0 before a dense layer with LeakyReLU activation reduces the data dimensionality to 64 before a final 5% dropout layer. The final dense layer produces five outputs, which are trained to predict the fluorescence intensity values for the five mamu alleles. The model is compiled with the Adam optimizer[30] and uses mean squared error (MSE) loss.

Hyperparameter search

Dropout, batch size, and the number of epochs were optimized using the hyperas wrapper for hyperopt[31]. The hyperparameter search allowed a uniform range between 0 and 0.6 for each of the three dropout layers. The search for epoch and batch size hyperparameters had binary choices. Epochs were either 1,000 or 2,000. Batch size was either 2,500 or 5,000. The hyperparameter search was run with the tree of parzen estimators algorithm[32] allowing a maximum of 100 evaluations. The optimal parameters from this search are in **Table 1**.

Table 1: Optimal hyperparameters for deep learning model. Hyperopt was used to determine the ideal hyperparameters for the model using a tree of parzen estimators algorithm over 100 evaluations.

Dropout Layer 1 Rate	0.44
Dropout Layer 2 Rate	0.2
Dropout Layer 3 Rate	0.05
Batch Size	5000
Epochs	1000

Final Model Training

The final model was re-trained using the best hyperparameters from **table 1**. Loss (as MSE) for training and validation data was plotted against the training epochs to monitor overfitting (**Supplemental Figure 3**).

Model Performance - Regression Metrics

Test peptides were input to the final trained model and the predicted intensities were compared with the experimental intensities**.** Correlations between true and predicted values were assessed by MSE, Spearman's rank correlation coefficient (Spearman's ρ), and the correlation p-value.

Positional SHAP

Shapely Additive Explanations (SHAP)[14] were used to determine the contribution of each position on each peptide to the peptide's overall predicted intensity. Training peptide sequence data was summarized as 100 weighted samples using the SHAP kmeans method. The summarized data, the test peptide sequence data, and the trained model were input into SHAP's KernelExplainer method. The contribution of each amino acid at each position was stored in an array. The mean SHAP value of each amino acid at each position was calculated for each mamu allele**.** Exemplary plots of the top predicted peptides were generated using SHAP's force_plot method indexed with peptides and position[15]. Dependence plots were generated using SHAP's dependence_plot method and modified with MatPlotLib[33].

Data and Code Availability

The data and code are available at https://github.com/jessegmeyerlab/positional-SHAP .

RESULTS

Model Training and Prediction

Peptide array data from Haj *et al.*[27] was composed of peptide sequences and their corresponding peptide array signal intensities, which are a proxy for MHC allele binding strength. Each peptide in the table had values for five Mamu MHC class I alleles: A001, A002, A008, B008, and B017 (**Figure 1**). Data was split into training, validation and test sets in a manner that ensures truncated versions of the same core peptide are in the same set (**Supplemental Figure 1**). The LSTM model used peptide sequences converted to integers as input to an embedding layer and learned to perform multi-output regression to outputs of the five Mamu MHC alleles as output (**Figure 1, Supplemental Figure 2).** Finally, the key contribution of this study is the concept of "positional SHAP" for sequence-dependent model interpretation.

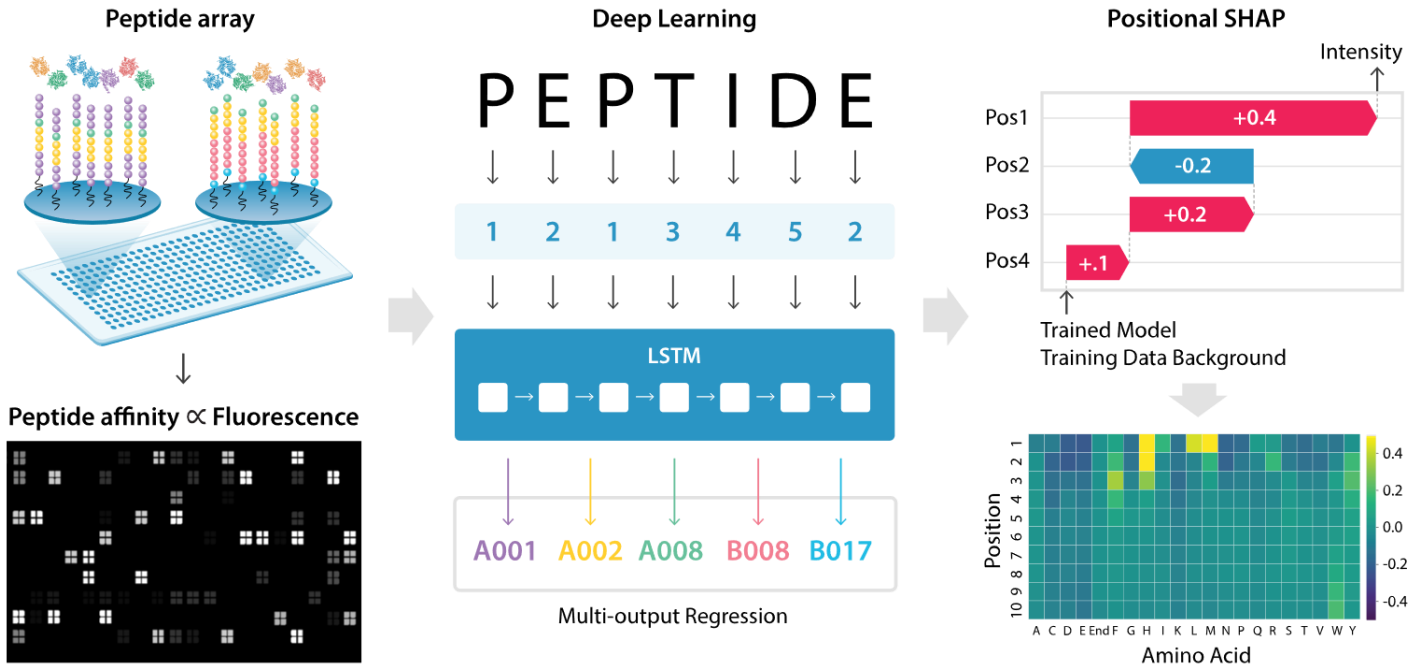


Figure 1: Overview of data, modeling, and positional SHAP analysis for model interpretation. Data downloaded from Haj *et al.* 2020 was generated by an immunofluorescence peptide array assay where each peptide's reported intensity is proportional to the amount of each MHC isoform that binds that peptide. The peptide sequence was numerically encoded and split to positional inputs, and a bidirectional Long Short-Term Memory (LSTM) model was trained to predict the five peptide array intensities, one for each MHC allele measured in the assay. The trained model was then used to make predictions on a separate test subset of the data. Finally, the model interpretation method SHAP was adapted to enable determination of each amino acid position's contribution to the final prediction. This positional SHAP analysis was visualized by plotting the mean SHAP value of each amino acid at each position as a heatmap.

The LSTM model easily achieved excellent performance on this multi-output regression task. Training versus validation loss for the final model (**Supplemental Figure 3**) demonstrates some overfitting but not to the detriment of the model's generalizability. Plots of true peptide intensity versus model predictions for the true held-out test set showed excellent performance for all the five alleles according to MSE values, spearman correlation analysis (**Figure 2**). All correlations between true and predicted values were significant with p-values less than 1.0E-16.

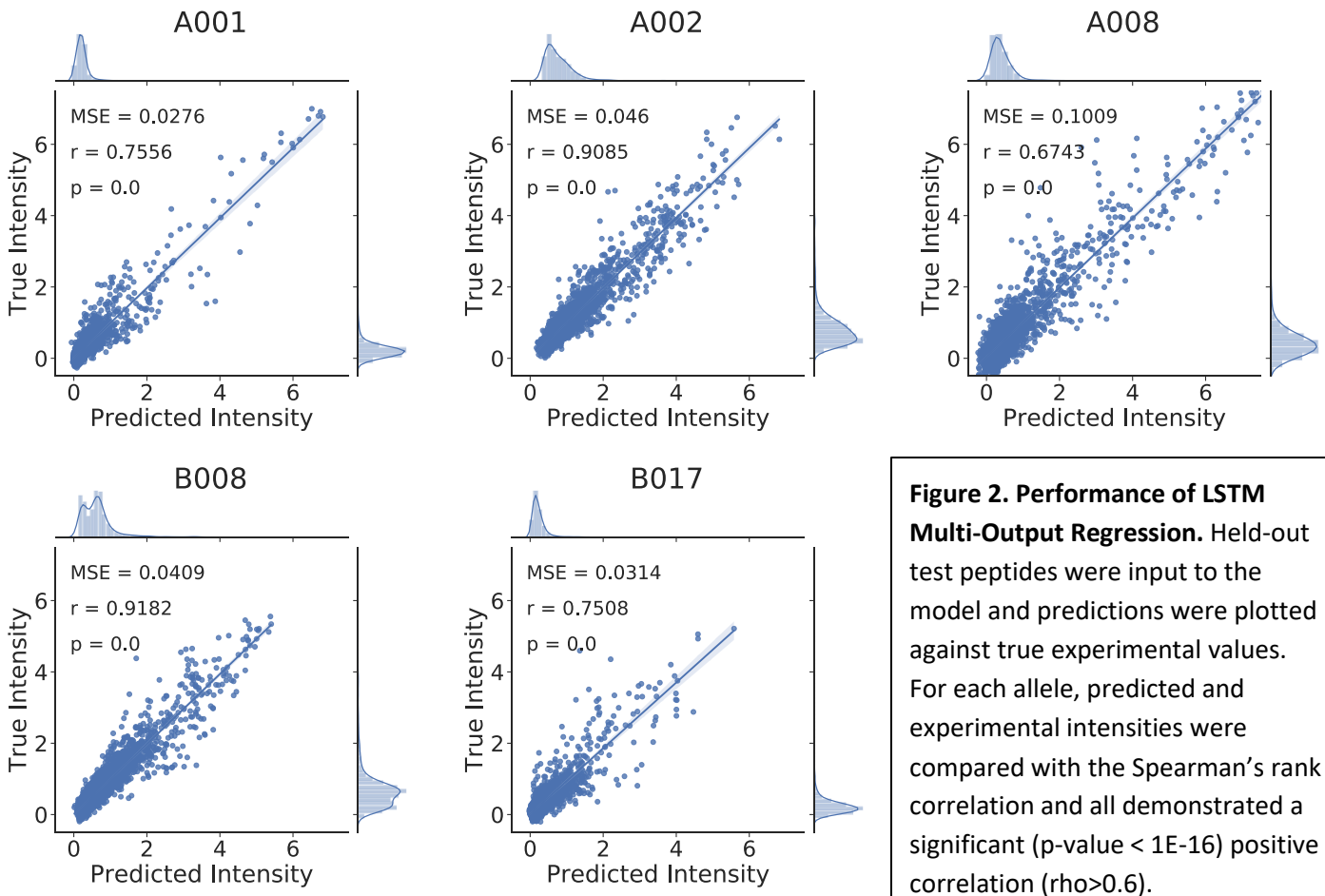


Figure 2. Performance of LSTM Multi-Output Regression. Held-out test peptides were input to the model and predictions were plotted against true experimental values. For each allele, predicted and experimental intensities were compared with the Spearman's rank correlation and all demonstrated a significant (p-value < 1E-16) positive correlation (rho>0.6).

Positional SHAP

Positional SHAP analysis revealed that each of the five alleles had different amino acid preferences that induced high binding/intensity predictions **(Figure 3)**. Mamu A001 generally prefers hydrophobic or hydroyl-containing amino acids in the first position (F, I, L, V, or Y), with a strong preference for S, T, or P in the second position; S and T are very similar chemically, with small, polar side chains containing hydroxyl functional groups. The heatmap of positional SHAP values also showed that A001 preferred proline between positions two and six. The preference of Mamu A002 was similar to A001 in that n-terminal serine or threonine result in high binding, but the preference for proline was completely absent. The preference map of mamu A008 showed an opposite trend, where only the preference for early proline between positions one and four is readily apparent and the contribution of S or T is absent. B008 appears to be most selective for the amino acids near the N-terminus, with a strong preference for arginine or methionine and strong negative SHAP values for many amino acids. Finally, B017 showed a preference for L, M, or H followed by F near the N-terminus. The heatmap of SHAP values for B017 also showed a positive contribution to binding from tryptophan near the C-terminus, suggesting that the entire peptide length may play a bigger role in binding to the B017 MHC protein.

Positional SHAP analysis also reveals the amino acids at each position that decrease peptide binding. All MHC alleles except for A002, and most pronounced in B008, have a strong negative contribution to binding prediction if there is an acidic amino acid in position one or two (i.e. D or E). For all alleles except B017, histidine near the peptide N-terminus also predicts low binding affinity.

Figure 3: Heatmaps showing positional SHAP analysis to determine amino acid binding motifs from deep learning models. The mean SHAP values for each amino acid at each position across all peptides in the test set were arranged into a heatmap. The position in each peptide is along the y-axis and the amino acid is given along the x-axis. "End" is used in positions 9 and 10 to enable inputs of peptides with length 8,9, or 10. For comparison, the SHAP force plot for the peptide with the highest binding prediction is shown for each allele.

Positional SHAP can further reveal the pooled binding contributions for arbitrary subsets of peptides. When the positional SHAP heatmap is filtered for the eight peptides with the highest binding predictions (top 0.013%), distinct patterns emerge **(Supplemental Figure 4)**. The serine or threonine at position two remains important for the A001 and A002. Proline emerges as important for B008 while proline at position three hinders binding to B017.

Inter-Positional Dependence

For each Mamu allele, at least one amino acid contributes to high binding prediction across multiple peptide positions. For example, the heatmap for MHC A001 **(Figure 3**) showed that proline positively contributes to binding when present in a wide range of positions from two to nine. Threonine positively contributed to binding at all positions for MHC allele A002. This suggested that each MHC binding pocket may tolerate their preferred motif with different absolute positions in the peptide, which prompted further investigation of the inter-positional dependence of peptide binding.

The SHAP value of any position is dependent on the values of all other positions in the peptide. SHAP dependence plots show how the range and variability of a position's SHAP value is reflective of the magnitude of its dependence on another position's amino acid value **(Figure 4).** For the A001 allele, position one has the greatest range of SHAP dependence values ranging from about -1 to +2. The amino acids with the greatest positive range at position one, for example, are serine, threonine, leucine, and isoleucine (**Figure 4A**), reflecting the results on the heatmap in **Figure 3**. Values with negative ranges are also reflected in the heatmap, including aspartate, glutamate, glycine, histidine, asparagine, and proline.

By referencing one position in the peptide given another position, we can tease out the interactions between neighbor amino acids that determine high peptide binding. For Mamu A001, serine and threonine are only assigned a high SHAP value when the amino acid at position two is a proline (dark pink, **Figure 4A**). Similarly, when the first amino acid is phenylalanine, the subsequent amino acid must be either a proline (dark pink) or a serine (light gray) at position two, but when the first amino acid is a different hydrophobic amino acid, leucine, a threonine is instead required (dark gold). Additionally, a similar but more subtle pattern is apparent for tyrosine and isoleucine; higher values are predicted when tyrosine is followed by a proline, and higher values predicted for isoleucine when followed by a threonine. Interestingly, these same amino acids at position two that predict high binding when starting with aliphatic amino acids result in lower values when a peptide starts with aspartic acid, glutamic acid, glycine, histidine, asparagine and proline. Thus, inter-positional SHAP analysis reveals relationships that are not apparent from the bulk heatmap analysis.

Beyond position one, a similar pattern is seen in relation to serine and threonine. At positions two, three, and four, the highest predictions for serine and threonine only result when they are followed by a proline (**Supplemental Figure 5**). Proline also shows a large positive range of SHAP values from position two through position five. However, there is variability even given an amino acid pair represented by these plots. The proline at position two, for example, has the highest prediction when it is followed by a glutamic acid (**Figure 4B**). However, there is also a point representing a peptide with proline followed by glutamic acid near the bottom of the range. Additionally, proline followed by glutamic acid is not the top prediction for any of the following positions. Beyond this, no simple pattern is apparent for amino acids that follow proline. The only predictor of a high SHAP value for proline is a preceding serine or threonine **(Supplemental Figure 6)**. These plots also reveal that phenylalanine and leucine only contribute to high binding affinity when at position one.

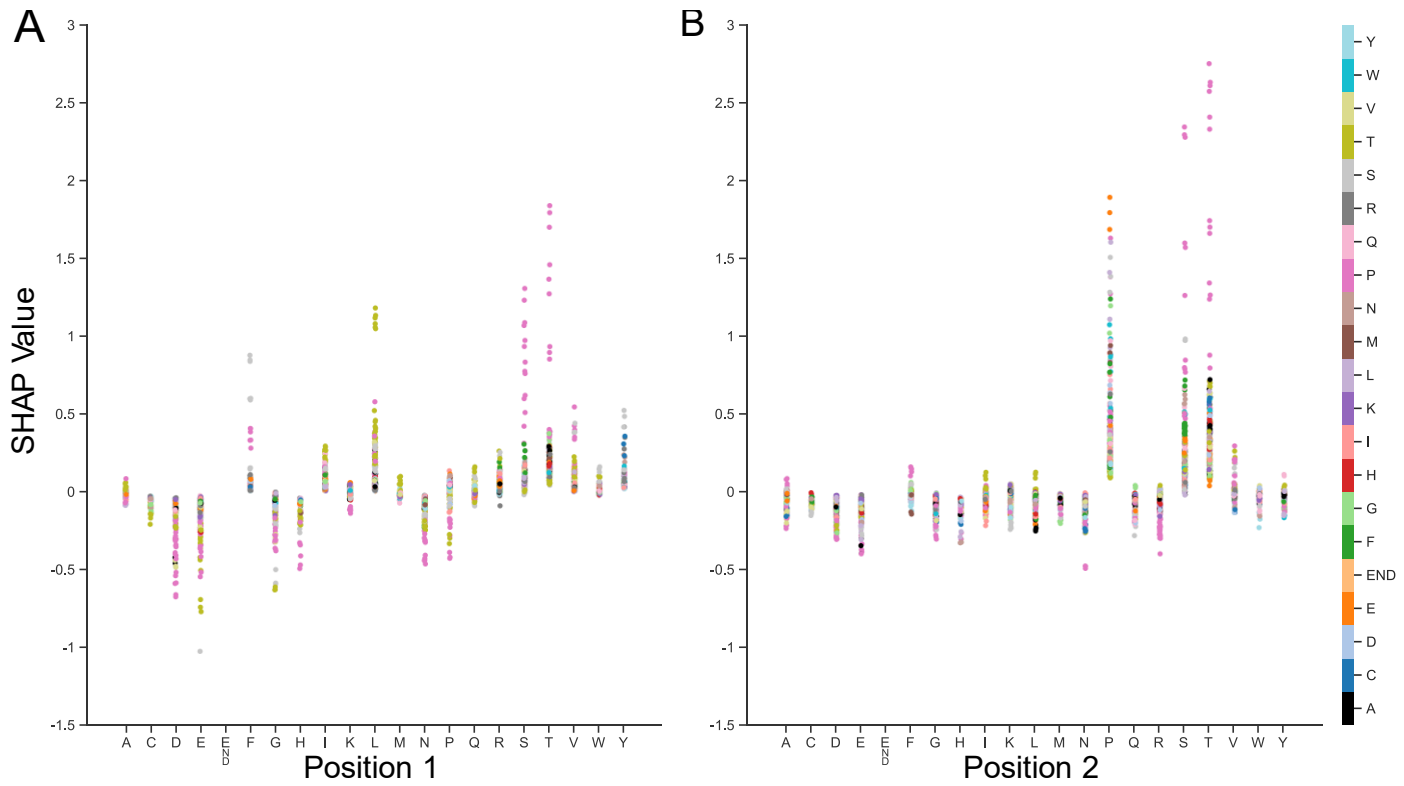


Figure 4. SHAP dependence plots were created for allele Mamu-A001 show how relationships between sequential amino acids contribute to binding. Each graph represents a pair of positions in the peptide, here positions one and two (A) and positions two and three (B). The x-axis lists each possible amino acid for that position and the y-axis shows the SHAP value. Each point represents a peptide with the listed amino acid at that position on the x-axis and the amino acid in the subsequent position is shown by color. This shows how the range of SHAP values for a particular amino acid at a specific location is reflective of the dependence of other amino acid positions.

Although interactions of adjacent positions are easily apparent, interactions between more distant positions can also be studied with this method. Most notable is the range of the “end” value at positions nine and ten, representing a peptide of less than ten positions. The values range through both negative and positive contributions from roughly -0.25 to 0.25. The positive values of “end” at position nine or ten are linked to a proline at position two (**Supplemental Figure 7**).

Finally, given the results presented, it may be useful to compare the patterns observed with top predicted peptides **(Figure 3)**. For the MHC allele A001, the top predicted peptide has a sequence of FSPPPAYVQQ. This matches the pattern displayed on the heatmap, with phenylalanine in position one, serine in position two, and three prolines spanning positions three to position five. The sequence also reflects the specificity noted from the interaction plots. Similar congruence between the top predicted peptides and the patterns displayed by the heatmap and interaction plots can be observed for the other four alleles **(Figure 3, Supplemental Figure 5)**.

DISCUSSION

The main contribution of this work is the concept of positional SHAP analysis to interpret neural networks trained from biological sequences. We show how positional SHAP can reveal amino acid motifs that influence MHC I binding, and further describe how positional SHAP enables understanding of inter-positional dependence of amino acids that result in high affinity predictions. This work also contributes a method for accurate prediction of peptide-MHC I affinity using peptide array data enabled by novel application of a neural network that combines amino acid embedding and LSTM layers.

Although there is an abundance of effective neural network models for biological sequences, there are a dearth of methods to understand those models. Thus, positional SHAP fills a gap in the biological machine learning community. Prior studies have used sequence logos from the top predictions[34], but this method does not ask the model what was learned and is instead sequence centric. One effective approach used by DeepLigand[25] is to apply Sufficient Input Subset (SIS) analysis[35], which attempts to reduce inputs to the minimal values required for prediction. Another recent report described the use of SHAP to interpret deep neural networks that predict the collisional cross section of peptides analyzed by ion mobility mass spectrometry[36], but without reference to the position of each amino acid, the utility of SHAP to understand position-specific insights from the model are limited.

By using positional SHAP to identify the contribution of each amino acid at each position to the final model's predictions, we can use the contributions to make hypotheses about the structure of the binding motif. For example, using SHAP independently, one would find that proline is an important amino acid to the prediction of a high binding affinity for A001. With positional SHAP, we can see that it is important for proline to be between positions two and five **(Figure 3)**, suggesting that the complementary binding region of the MHC complex requires at least one additional amino acid at the N-terminal end to bind properly and that there is a limit to how far along towards the C-terminus of the peptide the proline must be to properly bind.

Furthermore, by analyzing the positional SHAP data with dependence plots, we can determine the interactions between amino acids and how their relationship affects the overall predicted binding. The results presented herein show how these reveal peptide motifs that are bound by the MHC complex. For example, using only the heatmap analysis of A001, the highest predicted peptides are those with a phenylalanine or a leucine at position one, followed with either a serine or threonine at position two, and a proline at position three **(Supplemental Figure 4)**. The dependence plots for positions one compared to two reveal additional granularity **(Figure 4A)**; the highest SHAP values for phenylalanine are reserved for the instances where it is followed by a serine and for leucine where it is followed by a threonine. A similar pattern is apparent for tyrosine and isoleucine followed by serine and threonine, respectively. Thus, high binding affinity is dependent on an aromatic amino acid followed by S, or an aliphatic amino acid followed by T, such that the ideal binding sequences contain a core motif of either F/Y-S-P-X or L/I-T-P-X.

As an external validation of our results, we compared the binding preference of MAMU A001 determined from positional SHAP to a prior publication that determined specificity experimentally with a library of peptides with single amino acid substitutions[37]. This previous study determined a preference for "...S or T in position 2, P in position 3, and hydrophobic or aromatic residues at the C terminus". Our heatmap shows a similar preference (**Figure 3**), but we also note that F/I/L is preferred at position 1, and a proline at one of the positions between 2-5. The preference for a hydrophobic amino acid in position 1 was also seen using a substitution array in the original publication of the peptide array data used to train our models.[27]

Altogether the advances described herein are likely to find widespread use for both interpreting models trained from biological sequences, and for improving prediction performance using advanced model architectures.

Author Contributions

Conceptualization, QD, JGM; Methodology, QD, JGM; Software, QD, JGM; Validation, QD, JGM; Formal Analysis, QD, JGM; Investigation, QD, JGM; Resources, JGM; Data Curation, QD, JGM; Writing - Original Draft, QD, JGM; Writing - Reviewing and Editing, QD, JGM; Visualization, QD, JGM; Supervision, JGM; Project Administration, JGM; Funding Acquisition, JGM.

Acknowledgements

We thank Dasom Hwang for help with graphic design. This work was partially supported by T15 LM007359.

References

(1) Wüthrich, K. Protein Structure Determination in Solution by NMR Spectroscopy. *J. Biol. Chem.* **1990**, *265* (36), 22059–22062. https://doi.org/10.1016/S0021-9258(18)45665-7.

(2) Developments, applications, and prospects of cryo-electron microscopy - Benjin - 2020 - Protein Science - Wiley Online Library https://onlinelibrary.wiley.com/doi/10.1002/pro.3805 (accessed Mar 2, 2021).

(3) Protein crystallography from the perspective of technology developments: Crystallography Reviews: Vol 21, No 1-2 https://www.tandfonline.com/doi/abs/10.1080/0889311X.2014.973868 (accessed Mar 2, 2021).

(4) Huttlin, E. L.; Ting, L.; Bruckner, R. J.; Gebreab, F.; Gygi, M. P.; Szpyt, J.; Tam, S.; Zarraga, G.; Colby, G.; Baltier, K.; Dong, R.; Guarani, V.; Vaites, L. P.; Ordureau, A.; Rad, R.; Erickson, B. K.; Wühr, M.; Chick, J.; Zhai, B.; Kolippakkam, D.; Mintseris, J.; Obar, R. A.; Harris, T.; Artavanis-Tsakonas, S.; Sowa, M. E.; De Camilli, P.; Paulo, J. A.; Harper, J. W.; Gygi, S. P. The BioPlex Network: A Systematic Exploration of the Human Interactome. *Cell* **2015**, *162* (2), 425–440. https://doi.org/10.1016/j.cell.2015.06.043.

(5) Lee, J.; Freddolino, P. L.; Zhang, Y. Ab Initio Protein Structure Prediction. In *From Protein Structure to Function with Bioinformatics*; J. Rigden, D., Ed.; Springer Netherlands: Dordrecht, 2017; pp 3–35. https://doi.org/10.1007/978-94-024-1069-3_1.

(6) Kuhlman, B.; Bradley, P. Advances in Protein Structure Prediction and Design. *Nat. Rev. Mol. Cell Biol.* **2019**, *20* (11), 681–697. https://doi.org/10.1038/s41580-019-0163-x.

(7) Senior, A. W.; Evans, R.; Jumper, J.; Kirkpatrick, J.; Sifre, L.; Green, T.; Qin, C.; Žídek, A.; Nelson, A. W. R.; Bridgland, A.; Penedones, H.; Petersen, S.; Simonyan, K.; Crossan, S.; Kohli, P.; Jones, D. T.; Silver, D.; Kavukcuoglu, K.; Hassabis, D. Improved Protein Structure Prediction Using Potentials from Deep Learning. *Nature* **2020**, *577* (7792), 706–710. https://doi.org/10.1038/s41586-019-1923-7.

(8) LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521* (7553), 436–444. https://doi.org/10.1038/nature14539.

(9) Fukushima, K. Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. *Biol. Cybern.* **1980**, *36* (4), 193–202. https://doi.org/10.1007/BF00344251.

(10) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. Learning Representations by Back-Propagating Errors. *Nature* **1986**, *323* (6088), 533–536. https://doi.org/10.1038/323533a0.

(11) Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9* (8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

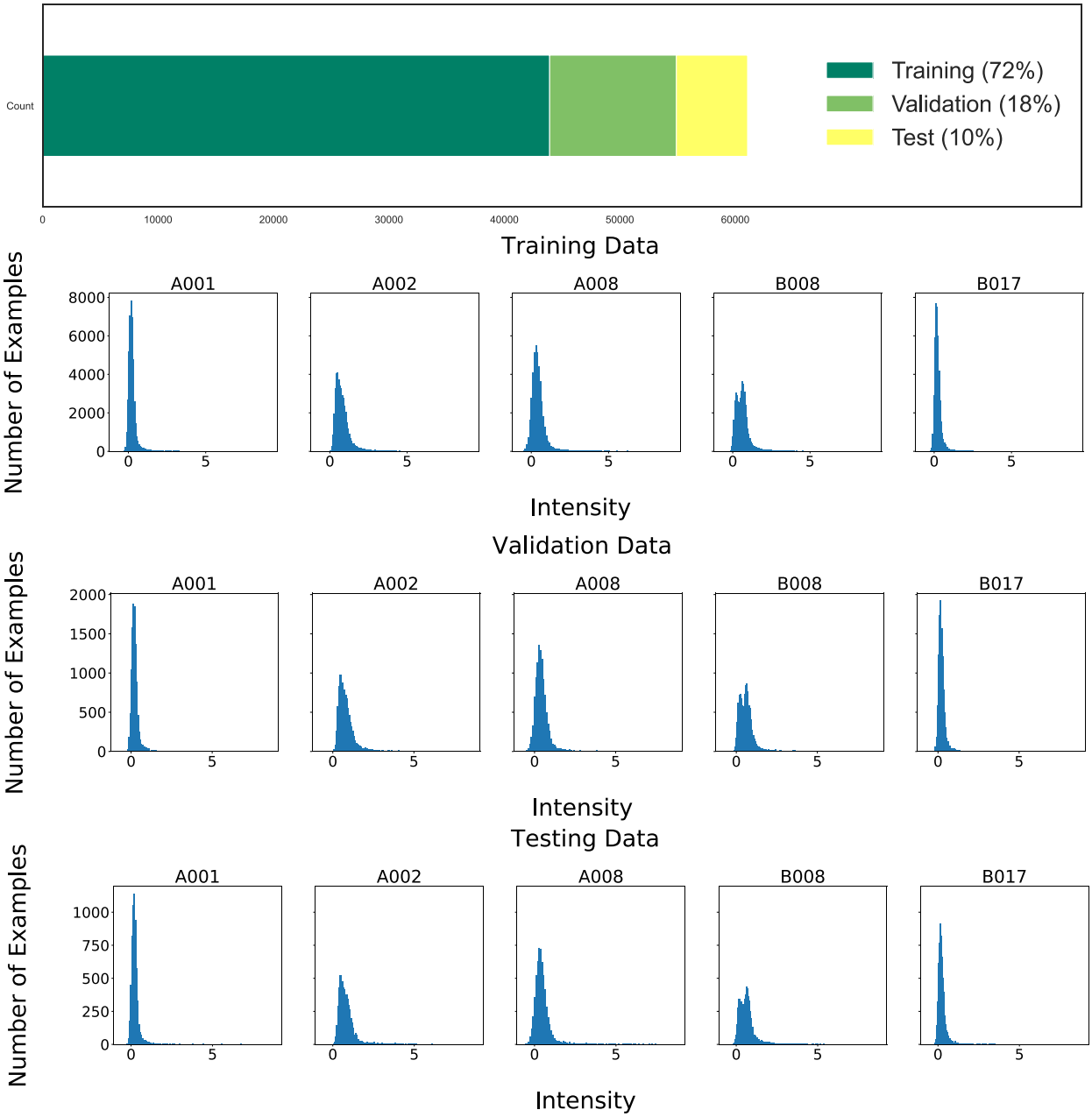
(12) Arras, L.; Arjona-Medina, J.; Widrich, M.; Montavon, G.; Gillhofer, M.; Müller, K.-R.; Hochreiter, S.; Samek, W. Explaining and Interpreting LSTMs. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*; Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., Müller, K.-R., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2019; pp 211–238. https://doi.org/10.1007/978-3-030-28954-6_11.

(13) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45* (1), 5–32. https://doi.org/10.1023/A:1010933404324.

(14) Lundberg, S. M.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*; NIPS'17; Curran Associates

Inc.: Red Hook, NY, USA, 2017; pp 4768–4777.
(15) Lundberg, S. M.; Nair, B.; Vavilala, M. S.; Horibe, M.; Eisses, M. J.; Adams, T.; Liston, D. E.; Low, D. K.-W.; Newman, S.-F.; Kim, J.; Lee, S.-I. Explainable Machine-Learning Predictions for the Prevention of Hypoxaemia during Surgery. *Nat. Biomed. Eng.* **2018**, *2* (10), 749–760. https://doi.org/10.1038/s41551-018-0304-0.
(16) Cresswell, P.; Bangia, N.; Dick, T.; Diedrich, G. The Nature of the MHC Class I Peptide Loading Complex. *Immunol. Rev.* **1999**, *172* (1), 21–28. https://doi.org/10.1111/j.1600-065X.1999.tb01353.x.
(17) 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.* **2011**, *11* (12), 823–836. https://doi.org/10.1038/nri3084.
(18) Matzaraki, V.; Kumar, V.; Wijmenga, C.; Zhernakova, A. The MHC Locus and Genetic Susceptibility to Autoimmune and Infectious Diseases. *Genome Biol.* **2017**, *18* (1), 76. https://doi.org/10.1186/s13059-017-1207-1.
(19) Sijts, E. J. A. M.; Kloetzel, P.-M. The Role of the Proteasome in the Generation of MHC Class I Ligands and Immune Responses. *Cell. Mol. Life Sci.* **2011**, *68* (9), 1491–1502. https://doi.org/10.1007/s00018-011-0657-y.
(20) Burrows, S. R.; Rossjohn, J.; McCluskey, J. Have We Cut Ourselves Too Short in Mapping CTL Epitopes? *Trends Immunol.* **2006**, *27* (1), 11–16. https://doi.org/10.1016/j.it.2005.11.001.
(21) Bonnal, S. C.; López-Oreja, I.; Valcárcel, J. Roles and Mechanisms of Alternative Splicing in Cancer — Implications for Care. *Nat. Rev. Clin. Oncol.* **2020**, *17* (8), 457–474. https://doi.org/10.1038/s41571-020-0350-x.
(22) Nielsen, M.; Andreatta, M.; Peters, B.; Buus, S. Immunoinformatics: Predicting Peptide–MHC Binding. *Annu. Rev. Biomed. Data Sci.* **2020**, *3* (1), 191–215. https://doi.org/10.1146/annurev-biodatasci-021920-100259.
(23) Antunes, D. A.; Abella, J. R.; Devaurs, D.; Rigo, M. M.; Kavraki, L. E. Structure-Based Methods for Binding Mode and Binding Affinity Prediction for Peptide-MHC Complexes. *Curr. Top. Med. Chem.* **2018**, *18* (26), 2239–2255. https://doi.org/10.2174/1568026619666181224101744.
(24) O'Donnell, T. J.; Rubinsteyn, A.; Bonsack, M.; Riemer, A. B.; Laserson, U.; Hammerbacher, J. MHCflurry: Open-Source Class I MHC Binding Affinity Prediction. *Cell Syst.* **2018**, *7* (1), 129-132.e4. https://doi.org/10.1016/j.cels.2018.05.014.
(25) Zeng, H.; Gifford, D. K. DeepLigand: Accurate Prediction of MHC Class I Ligands Using Peptide Embedding. *Bioinformatics* **2019**, *35* (14), i278–i283. https://doi.org/10.1093/bioinformatics/btz330.
(26) Liu, Z.; Cui, Y.; Xiong, Z.; Nasiri, A.; Zhang, A.; Hu, J. DeepSeqPan, a Novel Deep Convolutional Neural Network Model for Pan-Specific Class I HLA-Peptide Binding Affinity Prediction. *Sci. Rep.* **2019**, *9* (1), 794. https://doi.org/10.1038/s41598-018-37214-1.
(27) Haj, A. K.; Breitbach, M. E.; Baker, D. A.; Mohns, M. S.; Moreno, G. K.; Wilson, N. A.; Lyamichev, V.; Patel, J.; Weisgrau, K. L.; Dudley, D. M.; O'Connor, D. H. High-Throughput Identification of MHC Class I Binding Peptides Using an Ultradense Peptide Array. *J. Immunol. Baltim. Md 1950* **2020**, *204* (6), 1689–1696. https://doi.org/10.4049/jimmunol.1900889.
(28) Chollet, François. Keras https://keras.io/ (accessed Jan 12, 2021).
(29) Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mane, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viegas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; Zheng, X. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 19.
(30) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. *ArXiv14126980 Cs* **2017**.
(31) Bergstra, J.; Yamins, D.; Cox, D. D. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*; ICML'13; JMLR.org: Atlanta, GA, USA, 2013; p I-115-I–123.
(32) Bergstra, J. S.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for Hyper-Parameter Optimization. 9.
(33) Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* **2007**, *9* (3), 90–95. https://doi.org/10.1109/MCSE.2007.55.
(34) Jensen, K. K.; Andreatta, M.; Marcatili, P.; Buus, S.; Greenbaum, J. A.; Yan, Z.; Sette, A.; Peters, B.; Nielsen, M. Improved Methods for Predicting Peptide Binding Affinity to MHC Class II Molecules.

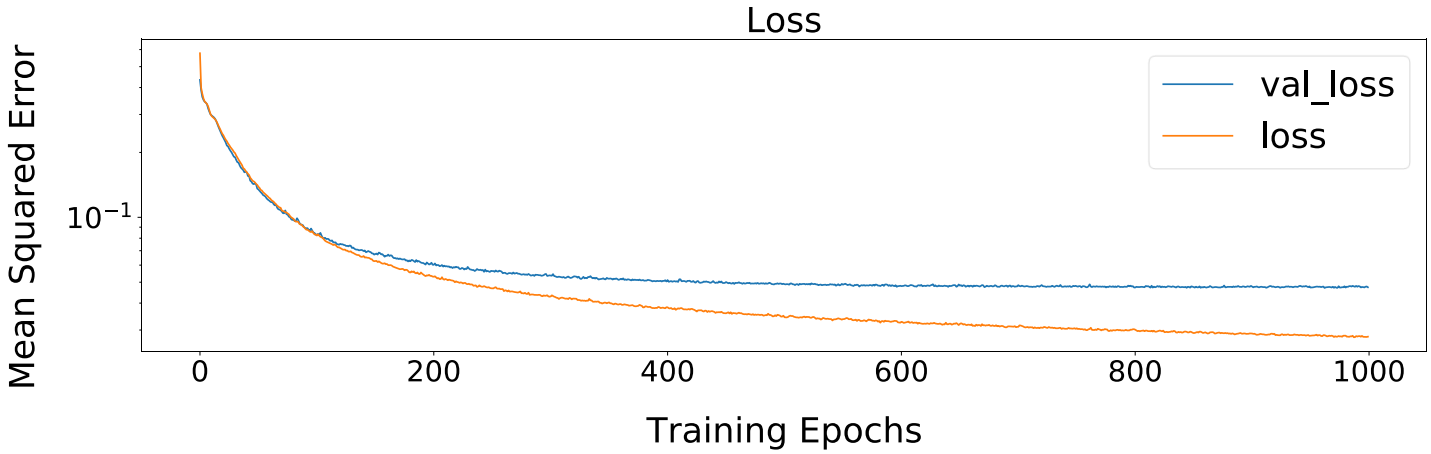
Immunology **2018**, *154* (3), 394–406. https://doi.org/10.1111/imm.12889.
(35) Carter, B.; Mueller, J.; Jain, S.; Gifford, D. What Made You Do This? Understanding Black-Box Decisions with Sufficient Input Subsets. *ArXiv181003805 Cs Stat* **2019**.
(36) Deep learning the collisional cross sections of the peptide universe from a million training samples | bioRxiv https://www.biorxiv.org/content/10.1101/2020.05.19.102285v1.full (accessed Jan 28, 2021).
(37) Sidney, J.; Dzuris, J. L.; Newman, M. J.; Johnson, R. P.; Amitinder, K.; Walker, C. M.; Appella, E.; Mothe, B.; Watkins, D. I.; Sette, A. Definition of the Mamu A*01 Peptide Binding Specificity: Application to the Identification of Wild-Type and Optimized Ligands from Simian Immunodeficiency Virus Regulatory Proteins. *J. Immunol.* **2000**, *165* (11), 6387–6399. https://doi.org/10.4049/jimmunol.165.11.6387.



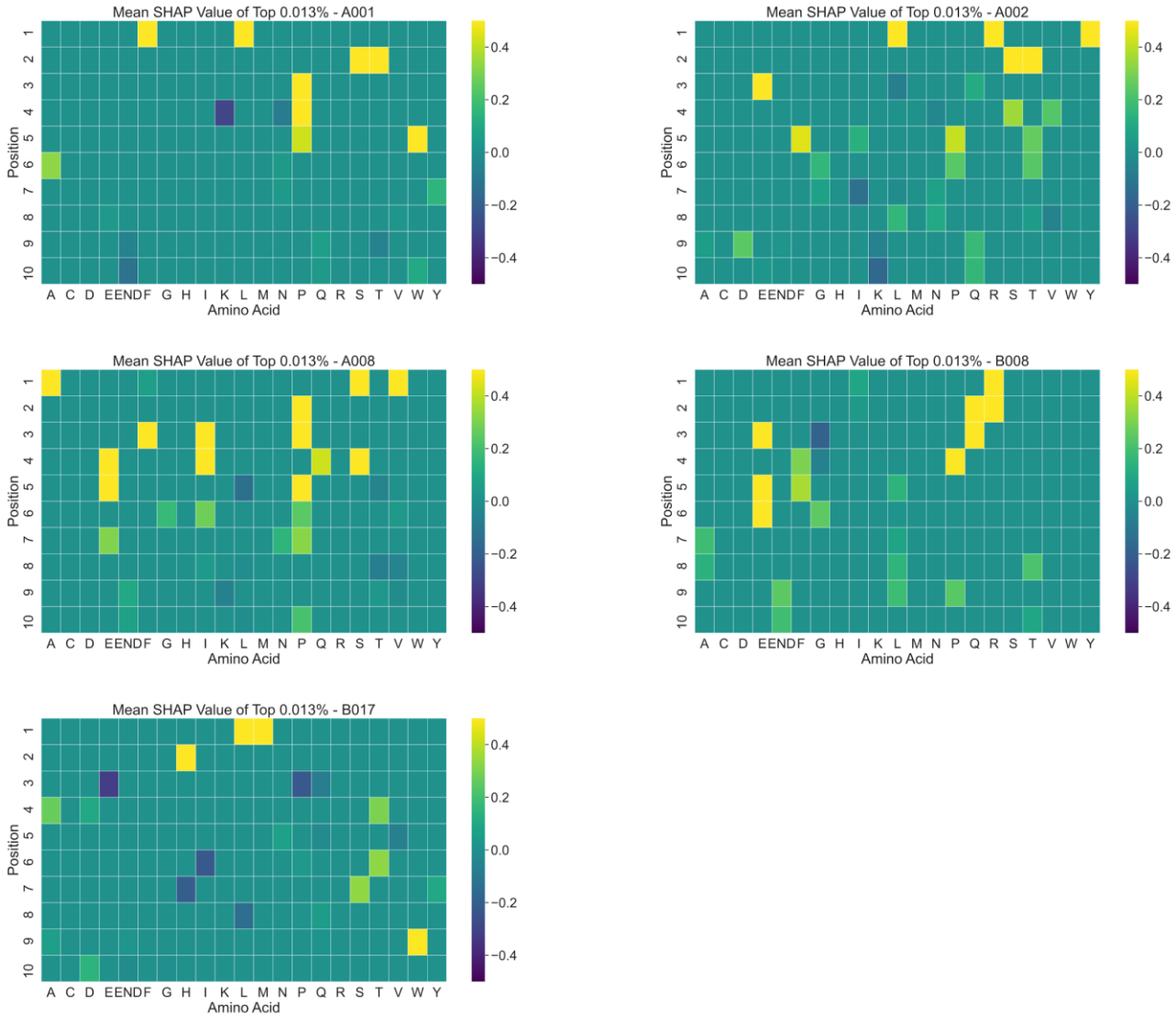
Supplemental Figure 1. Details of the data distributions and splitting. The data was split into three subsets. Training data made up 72% of the overall data and was used directly to train the model. Validation data made up 18% of the overall data and was used to monitor overfitting. Test data made up 10% of the total data and was used to test the final model's performance. The intensity distributions for each data subset were plotted for each allele to ensure that each maintained the same distribution.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 10, 50)            1050
_________________________________________________________________
lstm (LSTM)                  (None, 10, 128)           91648
_________________________________________________________________
dropout (Dropout)            (None, 10, 128)           0
_________________________________________________________________
lstm_1 (LSTM)                (None, 128)               131584
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense (Dense)                (None, 64)                8256
_________________________________________________________________
leaky_re_lu (LeakyReLU)      (None, 64)                0
_________________________________________________________________
dropout_2 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 5)                 325
=================================================================
Total params: 232,863
Trainable params: 232,863
Non-trainable params: 0
_________________________________________________________________
```

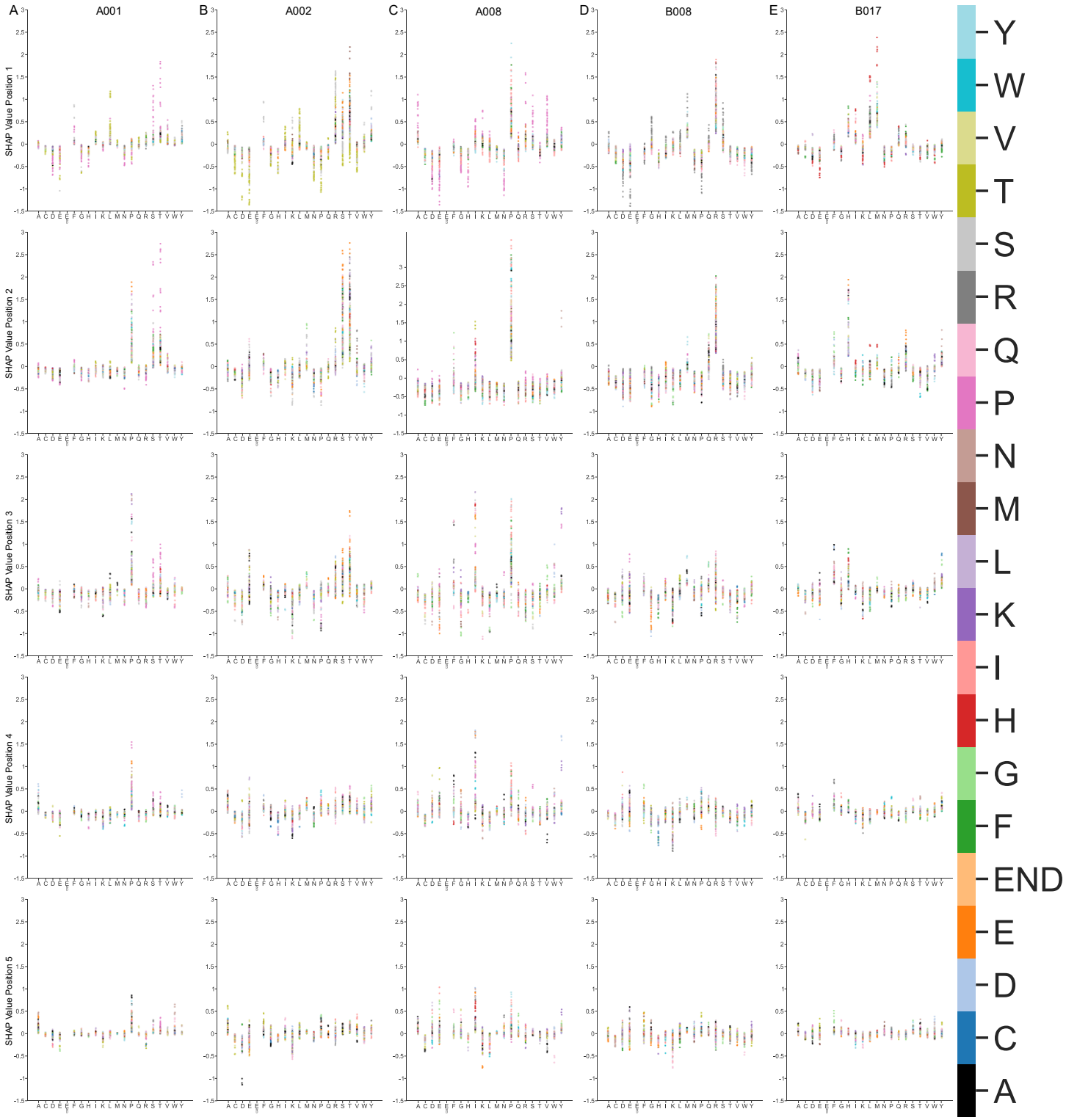
Supplemental Figure 2: Summary of LSTM model architecture. The architecture of the model consists of an embedding layer with 10 inputs with 21 dimensions each, representing each position of the peptide and each of the numeric representations of the possible amino acids and the end marker. This is followed by a pair of LSTM and dropout layers, with the dropout ratios determined by a hyperparameter search. Following the LSTM layers are a dense layer, a leaky ReLU activation layer, a final dropout layer, and a final dense layer with five outputs, each representing the intensity of the corresponding allele. The model was trained with a batch size of 5000 for 1000 epochs.



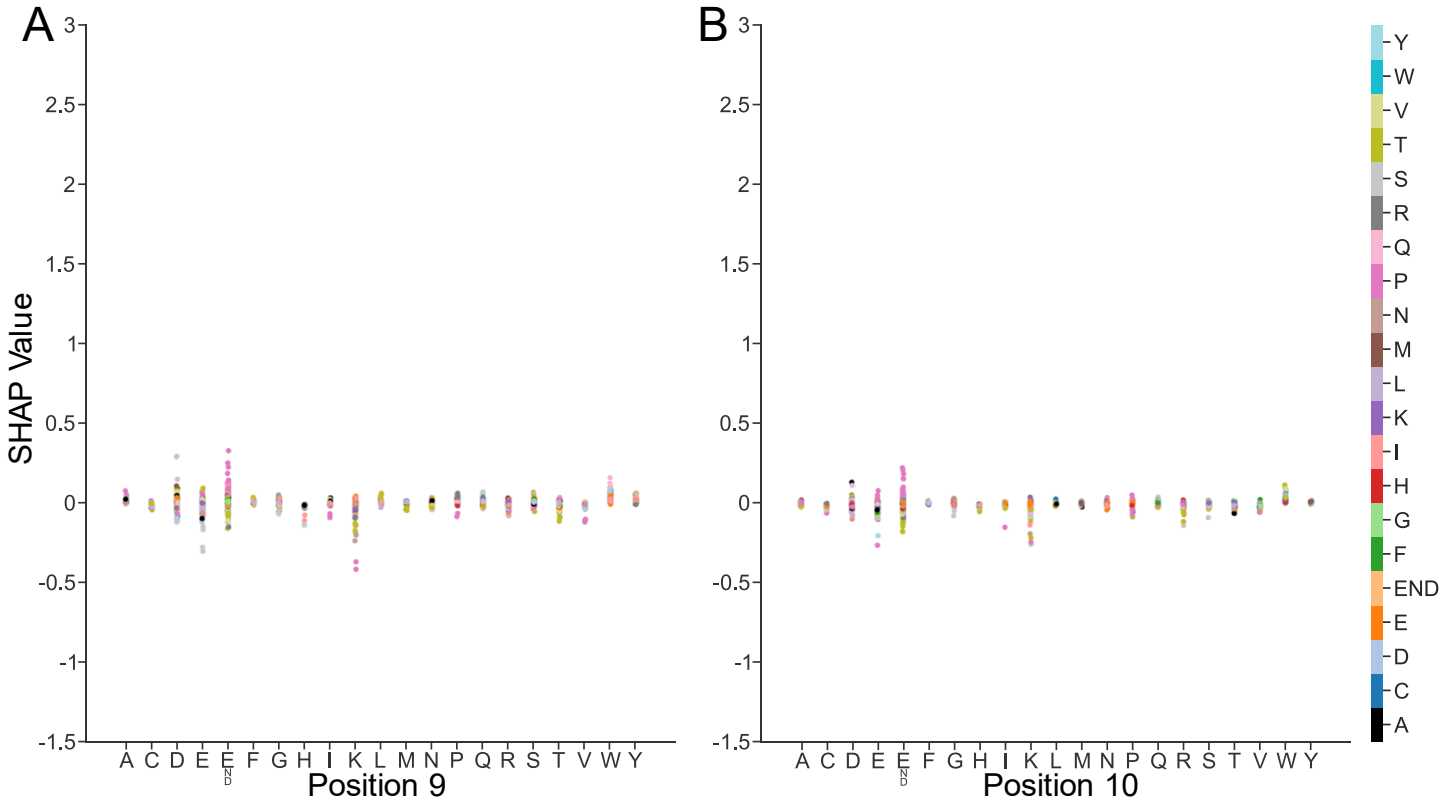
Supplemental Figure 3. Mean squared error loss over training. The model was trained for 1000 epochs and the loss from mean squared error between predictions and true, known values was plotted for both the training data and the validation data. The validation loss diverges from the test loss around epoch 175, indicating some amount of overfitting, however, the MSE of both datasets continues to decrease as the model is trained over 1000 epochs.



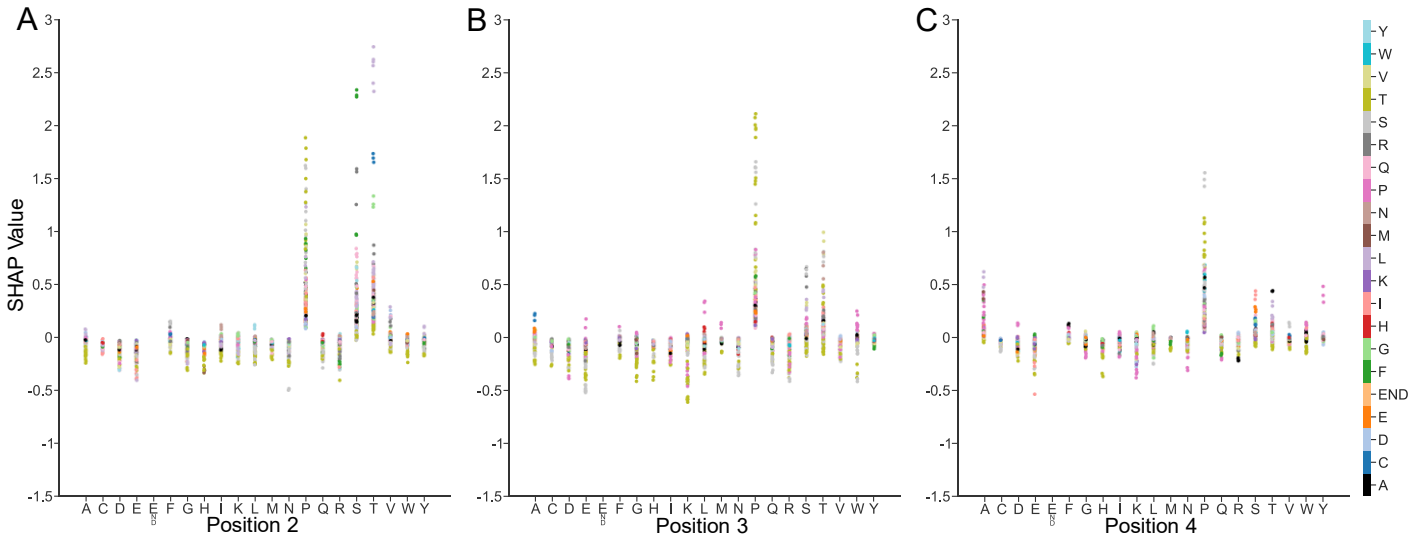
Supplemental Figure 4. Pooled positional SHAP for arbitrary subsets of the data. The mean SHAP values for each amino acid at each position were calculated for the peptides with the top 0.013% predicted intensity (top 8 peptides). Due to the small sample size, most of the amino acid positions have a value of zero. The positions with high values however, illustrate important amino acids for high intensity prediction. Phenylalanine or leucine are important at the first position for both A001 and A008. A serine or threonine at position two is important for A001, A002, and A008. All alleles, other than B017, demonstrate the importance of a proline near the middle of the peptide.



Supplemental Figure 5. Dependence Plots of First Five Positions for Each Allele. SHAP dependence plots were created for A001 (A), A002 (B), A008 (C), B008 (D), and B017 (E), for the first five positions. The colors represent the following position, e.g., the plot for position one has position two represented as colors. Each column represents each of the alleles, while each row represents the position.



Supplemental Figure 6. Dependence Plots of End Positions. SHAP dependence plots were created for allele A001, for positions nine and ten, with the colors representing position two of the peptide sequence. For the END value, representing a peptide less than ten amino acids, the assigned SHAP value is only positive when position two is a proline (dark pink).



Supplemental Figure 7. Dependence Plots for Preceding Position. SHAP dependence plots were created for allele A001, for positions two, three, and four, with the colors representing the preceding position, one, two and three, respectively. For each of the positions, proline has among the greatest range of SHAP values contributing to the predicted intensity. The highest SHAP values are only assigned when the preceding amino acid is either a serine (light gray) or a threonine (dark gold).