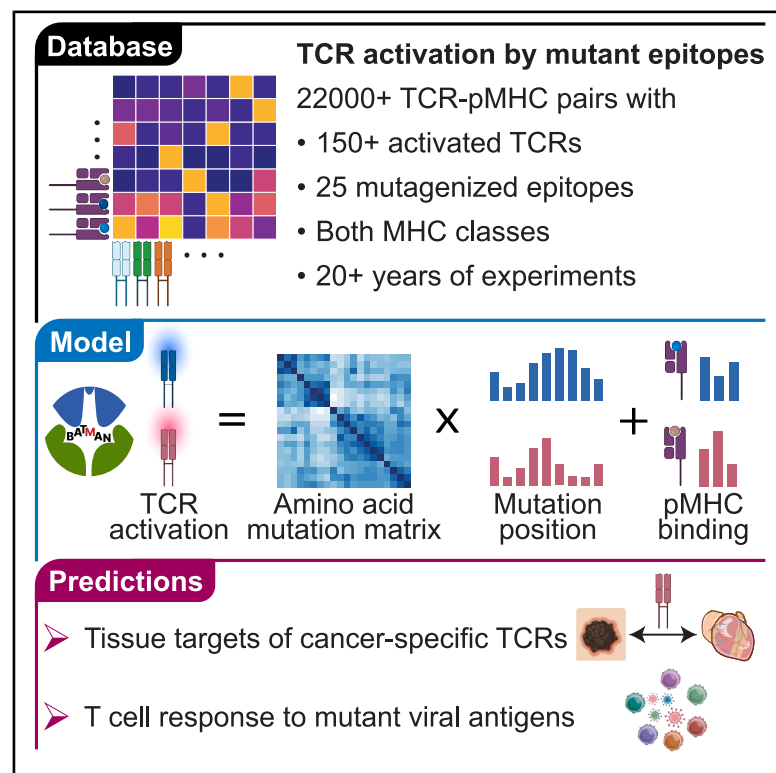


T cell receptor cross-reactivity prediction improved by a comprehensive mutational scan database

Graphical abstract



Authors

Amitava Banerjee, David J. Pattinson, Cornelia L. Wincek, ..., Sarah R. Chapin, Saket Navlakha, Hannah V. Meyer

Correspondence

navlakha@cshl.edu (S.N.),
hmeyer@cshl.edu (H.V.M.)

In brief

Banerjee et al. present a comprehensive T cell receptor cross-reactivity database and a hierarchical Bayesian framework to predict T cell receptor targets. Together, they reveal structural and biochemical features of T cell receptor epitope binding and identify therapeutic T cell receptor targets with high sequence dissimilarity.

Highlights

- BATCAVE is a T cell receptor cross-reactivity database of 22,000+ TCR-pMHC pairs
- BATMAN accurately predicts TCR cross-reactivity and ranks TCR targets
- Database and predictions reveal structural and biochemical features of TCR-pMHC binding
- BATMAN predicts therapeutic TCR targets with high sequence dissimilarity

Banerjee et al., 2025, Cell Systems 16, 101345

August 20, 2025 © 2025 Elsevier Inc. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

<https://doi.org/10.1016/j.cels.2025.101345>



Methods

T cell receptor cross-reactivity prediction improved by a comprehensive mutational scan database

Amitava Banerjee,¹ David J. Pattinson,² Cornelia L. Wincek,^{1,3} Paul Bunk,⁴ Armend Axhemi,^{5,6} Sarah R. Chapin,¹ Saket Navlakha,^{1,*} and Hannah V. Meyer^{1,7,*}

¹Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory, Cold Spring Harbor, NY 11724, USA

²Department of Pathobiological Sciences, School of Veterinary Medicine, University of Wisconsin-Madison, Madison, WI 53711, USA

³Medical Research Center and Clinic for Medical Oncology and Hematology, Cantonal Hospital St. Gallen, St. Gallen 9007, Switzerland

⁴School of Biological Sciences, Cold Spring Harbor Laboratory, Cold Spring Harbor, NY 11724, USA

⁵W.M. Keck Structural Biology Laboratory, Howard Hughes Medical Institute, New York, NY, USA

⁶Cold Spring Harbor Laboratory, Cold Spring Harbor, NY 11724, USA

⁷Lead contact

*Correspondence: navlakha@cshl.edu (S.N.), hvmeyer@cshl.edu (H.V.M.)

<https://doi.org/10.1016/j.cels.2025.101345>

SUMMARY

Comprehensively mapping all targets of a T cell receptor (TCR) is important for predicting pathogenic escape and off-target effects of TCR therapies. However, this mapping has been challenging due to lack of unbiased benchmarking datasets and computational methods sensitive to small-peptide mutations. To address this, we curated the benchmark for activation of T cells with cross-reactive avidity for epitopes (BATCAVE) database, encompassing near-complete single-amino-acid mutational assays, centered around 25 immunogenic epitopes, across both major histocompatibility complex classes, against 151 human and mouse TCRs, containing 22,000+ TCR-peptide pairs in total. We then introduce Bayesian inference of activation of TCR by mutant antigens (BATMAN), an interpretable Bayesian model, trained on BATCAVE, for predicting the peptides that activate a TCR, and an active learning extension, which efficiently maps targets of a novel TCR by selecting a few peptides to assay. We show that BATMAN outperforms existing methods, reveals structural and biochemical predictors of TCR-peptide interactions, and can predict polyclonal T cell responses and TCR targets with high sequence dissimilarity. A record of this paper's transparent peer review process is included in the supplemental information.

INTRODUCTION

A single T cell receptor (TCR) can recognize a variety of peptides, a property known as TCR cross-reactivity.^{1,2} Predicting which peptides a TCR cross-reacts to is critical for numerous applications, including predicting viral escape,³ cancer neoantigen immunogenicity,⁴ autoimmunity,^{2,5} and off-target toxicity of T cell-based therapies.^{6,7} However, predicting interactions among TCRs, peptides, and major histocompatibility complexes (TCR-pMHCs) remains challenging^{8–11} due to (1) limited TCR cross-reactivity assay data, (2) few experimentally validated negative examples,¹² which are important for model discrimination (Figure 1A), and (3) limited number of available ground-truth TCR-pMHC structures.¹³ Most existing computational methods are designed to cluster different TCRs that bind the same peptide.^{8,14} However, the opposite task—predicting peptides that bind a given TCR—remains outstanding.^{9,11,15,16} This is largely due to the sensitivity required to discriminate among single-amino-acid (aa) mutants^{11,17} of a TCR's known index peptide, i.e., the peptide to which the TCR was identified to strongly bind. To address this challenge, we offer

both a comprehensive experimental mutational scan database of TCR-pMHC binding and a method that can predict how peptide mutations affect TCR activation. Additionally, we also developed an active learning (AL) method to minimize the number of TCR-pMHC experiments that need to be performed to learn the peptide cross-reactivity of a novel TCR, thereby reducing costs associated with performing TCR-pMHC assays.

RESULTS

BATCAVE: A database of TCR-specific peptide mutational scans reveals TCR-pMHC interaction rules

To address the lack of TCR cross-reactivity training sets with balanced positive and negative experimentally validated TCR-pMHC pairs, we curated a database of continuous-valued TCR activation data by pMHCs from mutational scan assays (hereafter referred to as TCR activation data; Figures 1B and 1C). This database, termed BATCAVE (benchmark for activation of T cells with cross-reactive avidity for epitopes), includes (1) 35 fully sequenced CD8⁺ and 43 (7 fully sequenced) CD4⁺ mouse



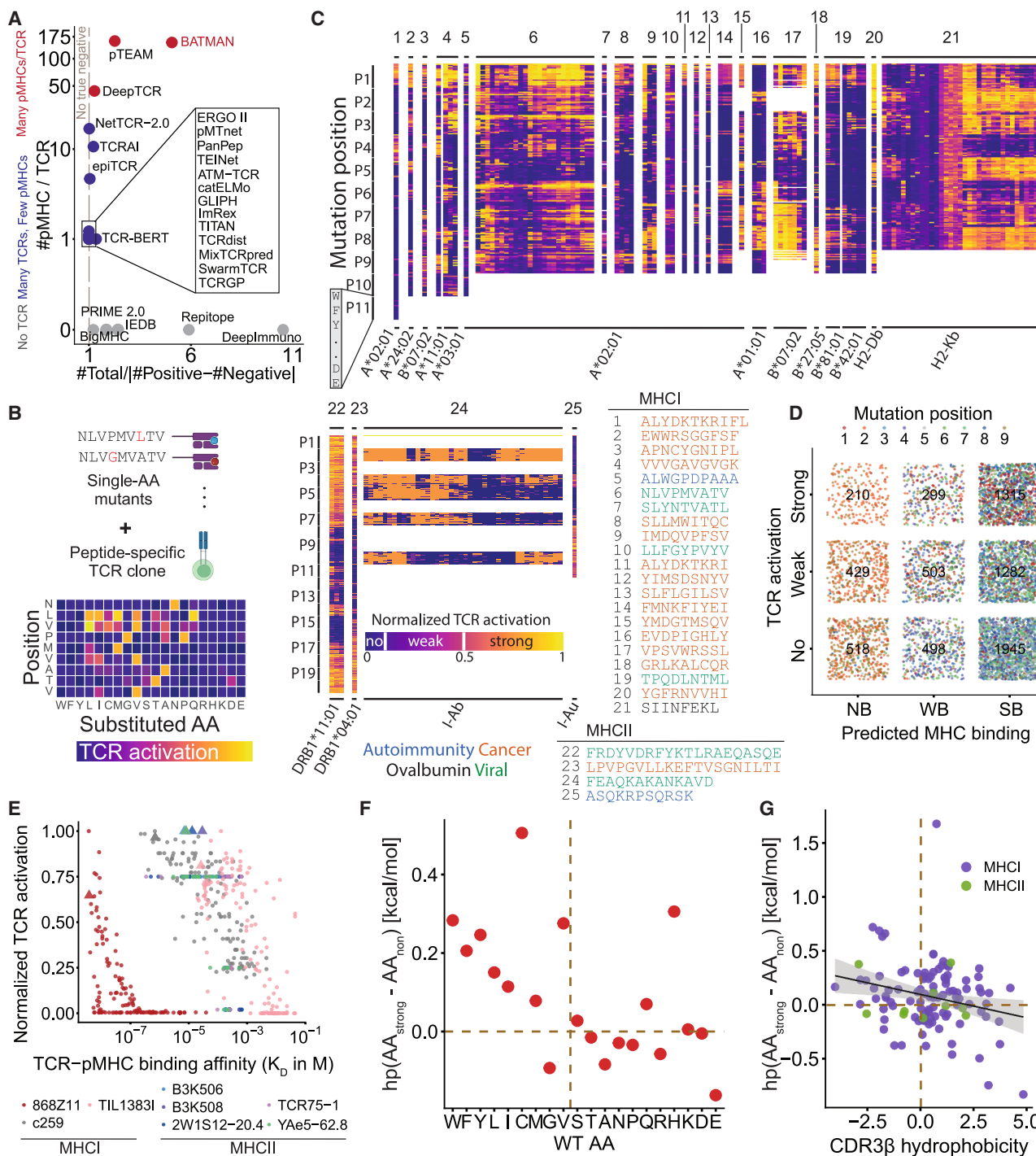


Figure 1. The BATCAVE database provides insights into TCR-pMHC interactions

(A) Training dataset summary metrics for the number of pMHCs tested per TCR and the class balance of experimentally verified TCR-pMHC interactions, shown for selected TCR-pMHC interaction prediction methods.

(B) Schematic of a mutational scan assay, reporting activation of a TCR clone against all single-aa mutants of its index (here, NLVPMVATV).

(C) Curated BATCAVE mutational scan database for TCR activation, with each column corresponding to a TCR clone, grouped by their index peptide (indicated above each column) and recognized MHC (below), and each row corresponding to the substituted aa at a specific position, ordered according to their interfacial hydrophobicity (linear unitless color scale).

(legend continued on next page)

TCR clones and (2) 69 (59 fully sequenced) CD8⁺ and 4 fully sequenced CD4⁺ human TCR clones (Figure S1). Together, TCRs in the BATCAVE recognize a total of 25 unique index peptides that are of length ($L \in [8, 20]$) aa presented by 11 class-I MHCs and 4 class-II MHCs molecules, which are involved in cancer, viral infection, or autoimmunity. For 101 of 151 TCRs, BATCAVE records the raw activation of all possible $L \times 19$ single-aa mutant peptides, and for the rest of TCRs, the corresponding data for all mutant peptides with mutations only in non-MHC-anchor positions (Figure 1C). To provide a consistent activation scale across diverse TCR-pMHC avidities and experimentally measured variables (e.g., interferon [IFN]- γ concentration and nuclear factor of activated T cells [NFAT]-GFP fluorescence intensity), we normalized the raw activation data to the maximum activation for each TCR. In addition to the TCR activation data, we collected matched TCR-pMHC dissociation constants of all single-aa mutant peptides for 3 MHC-I-restricted TCRs (868Z11, c259, and TIL1383I) and 5 MHC-II-restricted TCRs (all specific for the H2-IAb-bound “3K” antigen FEAQKAKANKAVD). Overall, we achieved (1) full coverage of the single-aa mutant antigenic space for over 100 TCRs and (2) balanced numbers of high-confidence, true-positive (TCR-activated) and true-negative (TCR-inactive) data for each TCR, both unprecedented among existing datasets (Figure 1A).

BATCAVE showed that single-aa changes of the index peptide can result in both loss and gain of TCR activation over orders of magnitude (Figures 1C and S2), and it recorded how different mutants of the same index peptide activate different TCRs,^{18–20} allowing us to distill broad, interpretable biochemical and cellular features of TCR-pMHC interactions. First, we analyzed how pMHC binding affects TCR activation. Since different MHC alleles have different peptide-binding anchor residues, we first focused on the most prevalent MHC allele in our database, HLA-A*02:01 (43 TCRs, 11 index peptides). We selected the 41 TCRs recognizing nine distinct 9-aa-long index peptides with all peptide positions mutagenized, and we used pMHC-binding-level prediction from NetMHCpan4.0²¹ to classify pMHC interaction as “no” (>2% rank), “weak” (0.5%–2% rank), or “strong” (<0.5% rank) binders. For ease of interpretation, we also discretized continuous TCR activation values into three levels (Figure 1C): strong ($\geq 50\%$ of maximum), weak (10%–50%), or no (<10%) activation. We found that the fraction of TCR strong-activating mutants increased from 0.18 for MHC non-binders, to 0.23 (weak-binders), and 0.29 (strong-binders) with predicted MHC binding. However, mutant peptides were distributed over all $3 \times 3 = 9$ possible paired (MHC-binding level $\times$ TCR activation level) categories (Figure 1D), with similar observations for other MHCs (Figure S3). This suggests that MHC binding alone cannot be used to predict TCR activation and demonstrates the need for TCR activation and peptide immunogenicity prediction models beyond pMHC affinity prediction.

Second, we utilized BATCAVE to deduce the relationship between TCR-pMHC binding (measured as K_d) and TCR activation.

This is a classic question in immunology with many features contributing to the relationship between binding and activation, such as pMHC concentration,²² TCR structural changes,^{23,24} catch bonds,^{25,26} and co-receptors.^{27–29} We found a general positive association between TCR-pMHC binding and TCR activation across both MHC-I and MHC-II TCRs (Figure 1E). However, there are exceptions; for example, for TCR 868Z11, which binds the disulfide-stabilized HLA-A02:01 molecule,³⁰ we found that strong TCR-pMHC binding of many mutants is not sufficient for TCR activation.

Third, we asked how TCR cross-reactivity depends on the wild-type (WT) aa of the index peptide. Since previous studies have associated hydrophobic peptide residues with increased immunogenicity,^{31–33} we asked whether WT aa residue hydrophobicity dictates the hydrophobicity of allowed mutations. To quantify this, we calculated the difference between average hydrophobicity (as measured by the Wimley-White interfacial hydrophobicity scale³⁴) of TCR non-activating and strong-activating mutated aa for each WT aa across all TCRs and peptide positions (Figure 1F). Positive differences in average hydrophobicity between strong-activating and non-activating mutated aa were more frequent when WT aa were hydrophobic (89%) than for any other class of WT aa (36%), with the mean differences in hydrophobic WT aa and all other aa being 0.18 and 0.00, respectively. This indicated that peptide positions with non-hydrophobic WT aa do not have any preference for the mutant aa for TCR activation, while hydrophobic WT aa residues preferentially allowed hydrophobic mutant aa, likely since the latter are crucial TCR-binding residues.

Finally, we studied how the TCR CDR3 chain hydrophobicity dictates the hydrophobicity and number of allowed mutations. For TCRs with available CDR3 α and CDR3 β sequences, we again calculated the difference between average hydrophobicity of TCR non-activating and TCR-activating mutated peptide aa. Next, we computed the hydrophobicity of the CDR3 $\alpha\beta$ chains by summing across the hydrophobicity of their individual aa residues. We found that hydrophobic (hydrophilic) CDR3 β chains prefer hydrophobic (hydrophilic) mutant aa for activation (Figure 1G), confirming a trend previously seen for TCR-pMHC structural analyses.^{32,35,36}

Overall, BATCAVE provides the largest to-date collection of TCR-pMHC interaction data across multiple MHC classes, index peptides, and TCRs and reveals broad as well as TCR-specific insights into properties of TCR-pMHC binding. Together, this database serves as a strong foundation for benchmarking TCR-pMHC prediction methods.

Existing TCR-pMHC models poorly predict TCR activation by mutant peptides

We first asked how well existing TCR-pMHC models predict TCR activation by mutant peptides, a question we could answer using BATCAVE as a benchmarking dataset. Since many index peptides appear multiple times for different TCRs in BATCAVE, we selected a subset of BATCAVE with each index peptide

(D) Dependency of mutant TCR activation on NetMHCpan-predicted pMHC binding, shown for all HLA-A*02:01-restricted 9-mer binding TCRs in BATCAVE, with the number of mutants (single dot) in each of the nine sub-categories (squares) indicated (total $n = 6,999$ dots across all 9 sub-categories).

(E) Dependency of TCR activation on TCR-pMHC binding, with dots representing TCR-pMHC pairs ($n = 560$) (index peptides indicated by triangles).

(F and G) Hydrophobicity difference between allowed and disallowed aa substitutions depend on WT peptide aa (F, dots representing average over all occurrences of a WT aa) and CDR3 β hydrophobicity (G, dots representing TCRs [$n = 114$]).

appearing only once. This selection yielded a diverse and balanced dataset, spanning 18 MHCII-restricted human TCRs specific for unique 9- and 10-aa-long index peptides and 4 MHCII-restricted TCRs specific for index peptides of various lengths (Figure 2A). For TCRs with the same index peptide, we included the one with the least class imbalance.

We benchmarked 15 pre-trained TCR-pMHC models with different machine learning architectures and input data requirements (Figure 2A), selected based on the availability of web-servers or source code. These models are trained on large publicly available databases (e.g., VDJdb,³⁷ McPAS-TCR,³⁸ and single-cell immune repertoire profiling data³⁹). While these methods are designed for either classifying TCRs binding to selected immunogenic peptides^{8,14,40} or predicting peptide immunogenicity,^{32,41} our benchmarking involved a distinct problem of classifying which set of peptides activate a given TCR. Similar to previous results on smaller and less diverse datasets,^{11,17} these models predicted only marginally better than random on our benchmarking dataset, with the best classification area under the curves (AUCs) being from TITAN⁴² (mean AUC = 0.56 for MHCII-restricted and 0.52 for MHCII-restricted peptides) and ImRex⁹ (mean AUC = 0.55 for MHCII-restricted peptides). General immunogenicity prediction tools, which do not use TCR sequence information for prediction, overall performed worse, with none included among the top 6 performers. Overall, predicted TCR-pMHC interaction scores from most tested TCR-pMHC models were uncorrelated with true TCR activation values for the mutant peptides (Figure S4). Thus, these results demanded development of improved TCR-pMHC models for predicting how peptide mutations affect TCR activation.

BATMAN: A Bayesian inference model provides state-of-the-art prediction of TCR activation by mutant peptides

We present BATMAN—“Bayesian inference of activation of TCR by mutant antigens”—a hierarchical Bayesian model that can predict TCR activation by single-aa mutant peptides based on their distances to the TCR’s index peptide. Our peptide-to-index distance is a product of (1) a learned positional weight profile, specific for individual TCRs, corresponding to the weighted effects of mutated residues at different positions in the sequence, (2) a learned aa substitution distance matrix from the index aa to the mutant aa, and (3) an optional scalar weight, also specific for individual TCRs, corresponding to the pMHC-binding category (Figure 2B; STAR Methods). BATMAN can be used for classification and continuous regression tasks for both TCR-specific and cross-TCR activation datasets.

We validated BATMAN over our benchmarking dataset in two modes (Figure 2C):

- (1) Within-TCR mode, where we performed 5-fold cross-validation for each TCR, with the folds stratified by TCR activation category and mutation position. The inferred positional weight profiles and pMHC-binding-level weights were TCR specific, while the inferred aa matrix was TCR independent. Even though we inferred TCR-specific parameters, the hierarchical Bayesian framework of BATMAN allowed pooling of training data across TCRs

to infer TCR-independent model hyperparameters and the aa matrix (see STAR Methods).

- (2) Leave-one-TCR-out (LOO-TCR) mode, where we trained on data from all TCRs but one and tested on the held-out TCR. The inferred positional weight profiles, pMHC-binding-level weights, and inferred aa matrix were all TCR independent and learned by training across all TCRs except the held-out one (see STAR Methods).

Since all of the pre-trained TCR-pMHC models (Figure 2A) performed close to random in our benchmarking study, we compared BATMAN’s performance with that of *pTEAM*,¹⁷ an existing method designed to predict the effect of peptide mutation effects on TCR activation. For a fair comparison, we re-trained *pTEAM* using the same data as BATMAN.

BATMAN outperformed *pTEAM* in both within-TCR (mean AUC = 0.836 versus *pTEAM* AUC = 0.795) and LOO-TCR (mean AUC = 0.687 versus *pTEAM* AUC = 0.613) classification (Figure 2D). Critical to achieving BATMAN’s performance was learning the aa distance matrices by pooling training data across TCRs. For example, applying BATMAN with the BLOSUM100 aa matrix, the best-performing conventionally used distance matrix (Figures S5–S7), dropped the within-TCR AUC to 0.811 (Figure 2D). Figures S5–S7 further highlight the superior performance of BATMAN over 68 conventional aa substitution distance matrices when tested on all 151 TCRs, both in classification and continuous regression tasks. BATMAN performance also slightly improved with the addition of pMHC-binding-level information, with the within-TCR AUC dropping to 0.833 without this input (Figure 2D). *pTEAM* can also incorporate TCR sequence information into its model, but we found no gain in performance with this additional information (Figure 2D), showing its limitations when the training data are sufficiently diverse.

How do BATMAN and *pTEAM* compare when training data become limited? We calculated the within-TCR AUCs from both models over random subsamples of previous training folds, maintaining stratification over TCR activation classes and mutation positions. In each case, we also trained within-TCR BATMAN in two ways: (1) pooled across all TCRs like before, and, to investigate effects of pooling across TCRs and the overall training data size, (2) trained and tested individually for each TCR. In this limited-data regime, BATMAN pooled across TCRs consistently performs better than *pTEAM*, with BATMAN achieving close to its 5-fold cross-validation performance with only a random 40% subsample of training data in each fold (7 per mutation position per TCR) (Figure 2E). Pooled BATMAN performed better than unpooled BATMAN, for which MHCII AUC dropped to 0.775 for the above 40% subsampling, though the difference narrowed with more training data (MHCII AUC = 0.830 and 0.817, respectively, for unpooled and pooled with full within-TCR data; see also Figure S7). Unpooled BATMAN also performed comparably to *pTEAM* in the very-low training data regime (Figure 2E). Overall, these results highlight the ability of BATMAN to improve TCR-pMHC interaction prediction by pooling across TCRs compared with *pTEAM*, which must be trained separately for every TCR.

Finally, we compared BATMAN and *pTEAM* for LOO-TCR tasks where the training data are restricted to TCRs with the same index peptide. For this benchmark, we selected 8 MHCII

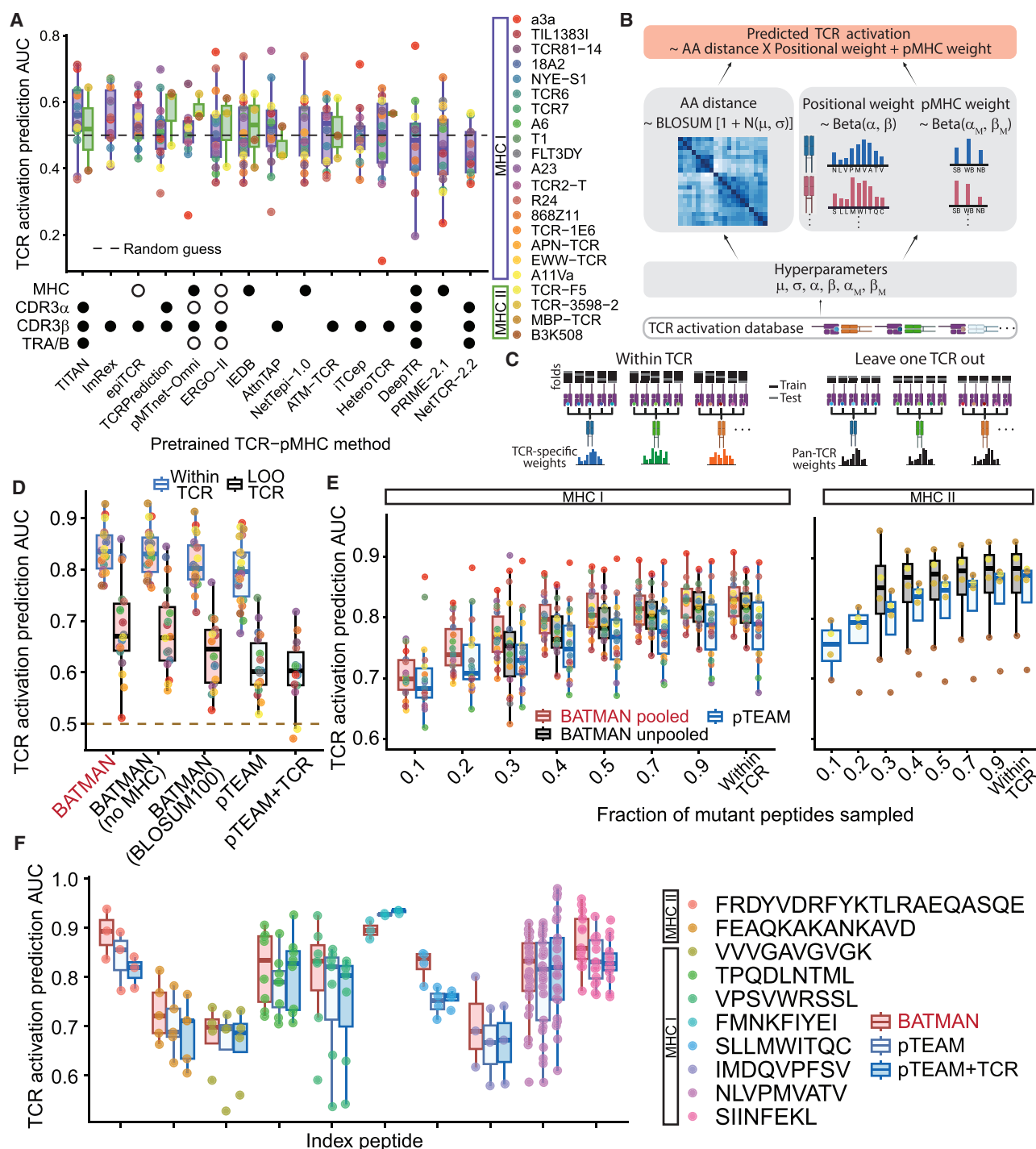


Figure 2. Batman outperforms existing TCR-pMHC interaction prediction methods

(A-C) (A) TCR activation classification AUC scores for different pre-trained methods, with their respective requirements indicated by dot matrix (filled: mandatory, open: optional, used if available in database); MHC: name of peptide-binding MHC allele; CDR3 α/β : sequence of the TCR CDR3 α/β chain; TRA/B: gene name or sequence of at least one of the TRAB, TRAJ, TRBV, and TRBJ gene name or sequence. Schematics for (B) Batman TCR activation prediction method and (C) two validation modes, within-TCR and LOO-TCR.

(D) BATMAN outperforms pTEAM in both within-TCR and LOO-TCR modes.

(E) BATMAN outperforms pTEAM for different numbers of training samples, showing improvement with pooling across TCRs (TCR color scheme same as in A).

(F) BATMAN and pTEAM LOO-TCR performance comparison among TCRs specific for a common index peptide.

and 2 MHCII index peptides for which mutational scan data against at least 3 TCRs were found in BATCAVE. BATMAN performed better than *pTEAM* for 9 out of 10 index peptides tested (Figure 2F), with their performances becoming comparable only for index peptides with a large number of TCRs tested (e.g., TPQDLNTML [$n = 8$], NLVPMVATV [$n = 25$], and SIINFELK [$n = 16$]). These benchmarking results collectively highlight the robust ability of BATMAN to predict TCR activation by mutant peptides across diverse TCR-pMHC training data contexts.

Inferred BATMAN parameters capture interpretable features of TCR-pMHC interactions

BATMAN inferred positional weights were consistent across different aa matrices (Figure S8), indicating that they might correspond to key TCR-pMHC interaction features. Thus, we asked what the inferred positional weight profiles and aa substitution matrices of BATMAN can reveal about TCR-pMHC interaction rules.

First, we inferred a pan-TCR positional weight profile and aa matrix by training BATMAN over all TCRs binding to 9-aa-long peptides in our benchmarking dataset of Figure 2A. The learned pan-TCR positional weights (Figure 3A) peaked near the middle of the peptide chain, reflecting the fact that central aa residues more directly affect TCR binding compared with flanking MHC-anchor residues^{19,41,43–46} (see also Figure S2).

Second, we pooled across TCRs but inferred TCR-specific weight profiles, which showed that while central residues are indeed often highly weighted, there can be variability of binding motifs for individual TCRs (Figures 3A and S8), suggesting that TCR cross-reactivity constitutes a diverse spectrum. Some TCRs (e.g., a3a and T1) are activated when very specific aa exist at certain positions (e.g., P2–P9 for T1 and P1–P5 for a3a), and by almost all aa at other positions (P1 and P6–P8, respectively). These TCRs have almost binary positional weight profiles and resulted in better within-TCR classification performance (Figure 2D). Similarly, positional weights could reveal how different TCRs that bind the same index peptide use different binding motifs. We trained BATMAN by pooling over all 25 BATCAVE TCRs specific for the NLV peptide and indeed found heterogeneity in positional weights across TCRs (Figure 3B). For example, the M5 residue is important for binding to most TCRs, while N1 and A7 residues show high variation in their importance across TCRs.

Third, we asked whether BATMAN can identify the exact TCR epitopes from mutational scans of longer MHCII-bound peptides, which would be necessary for T cell epitope discovery from peptidome-wide tiling scans (e.g., Dezfoulian et al.⁴⁷). To test this, we investigated BATMAN weights for the CD4⁺ TCRs TCR-F5 and TCR-3598-2 in the T-Scan-II dataset⁴⁷ performing mutational scans across all positions in 20-aa-long peptides. The inferred positional weights (Figure 3A) peaked at peptide positions included in the experimentally verified central TCR-binding epitopes, FRDYVDRFYKTLRAEQASQE.⁴⁸ and LPVPGVLLKEFTVSGNLT.⁴⁷ This suggests that BATMAN weights can reveal the exact TCR epitopes from longer peptide tiles with many TCR-irrelevant positions mutagenized.

Fourth, to study the correspondence between BATMAN positional weights and TCR-pMHC 3D structures quantitatively, we assessed the interactions between peptide residues and TCR

chains for 6 MHC I and 2 MHCII TCRs in BATCAVE with available ground-truth crystallographic structures of their corresponding TCR-pMHC complexes in the Protein Data Bank (PDB). BATMAN positional weights correlated strongly (r between 0.61 and 0.81) with the number of interactions between peptide residues and TCR chains (Figure 3C). For example, for the highly cross-reactive² autoimmune TCR 1E6 (PDB: 3UTT⁴⁹) binding the insulin peptide ALWGPDPAAA, positional weights are large for the solvent exposed GPD bulge of the peptide, over which the TCR binds.⁴⁹ Similar correspondence between large inferred positional weights and important TCR recognition motifs on peptides, characterized by a large number of TCR interactions, were seen for the other TCRs (Figure 3C). This suggests that BATMAN weights encapsulate key structural features of TCR-pMHC complexes that affect TCR-peptide binding.

Fifth, we investigated additional features of TCR-pMHC interactions revealed by the inferred positional weight profiles. As a test case, we used a dataset spanning a wide range of cross-reactivity, consisting of both low-avidity “naive” (broadly cross-reactive) and antigen-expanded “educated” (specific) TCRs binding to the SIINFELK peptide.¹⁸ We trained BATMAN by pooling over all TCRs and calculated the width of the posterior distribution of the inferred positional weights for each peptide residue which contains 90% of the density. To eliminate differences among TCRs due to their different pMHC-binding requirements, we focused only on the non-MHC-anchor positions (P1, P4, P6, and P7) of the SIINFELK peptide. We found that naive TCRs with broader cross-reactivity profile have broader posterior distributions of inferred weights (Figure 3D), suggesting that the weight profiles capture essential features of cross-reactivity.

Finally, the BATMAN aa distance matrix (Figure 3E) revealed three biochemical features of TCR-pMHC interactions. First, large changes in TCR activation correspond to non-aromatic to aromatic aa substitutions (e.g., valine and glycine to phenylalanine; methionine to tryptophan and tyrosine) affecting side-chain interactions.^{31,32,41} Second, swapping in hydrophobic isoleucine and leucine residues for non-hydrophobic residues overall increases TCR activation, in line with these residues considered to increase immunogenicity^{31–33} and our earlier observation in BATCAVE data (Figure 1F) that strong-activating mutant aa residues are more hydrophobic than non-activating residues for highly hydrophobic WT aa. Third, while previous works have reported both gain⁵⁰ and loss⁵¹ of TCR activation with peptide cysteine residue substitutions, we found an increase in TCR activation for valine, glutamine, and serine to cysteine substitution while there was a decrease in activation for phenylalanine and tyrosine to cysteine substitutions (Figure 3E), reflecting the high context dependence of such substitutions.

Overall, the interpretable parameters learned by BATMAN capture a host of biological features that reveal the nature of TCR-pMHC interactions.

Active learning with BATMAN boosts performance for novel TCRs

Designing cross-reactivity assays for a novel TCR is challenging because the number of possible peptides to test increases exponentially with the number of mutations. Can we identify a small set of informative peptides within this space, whose binding

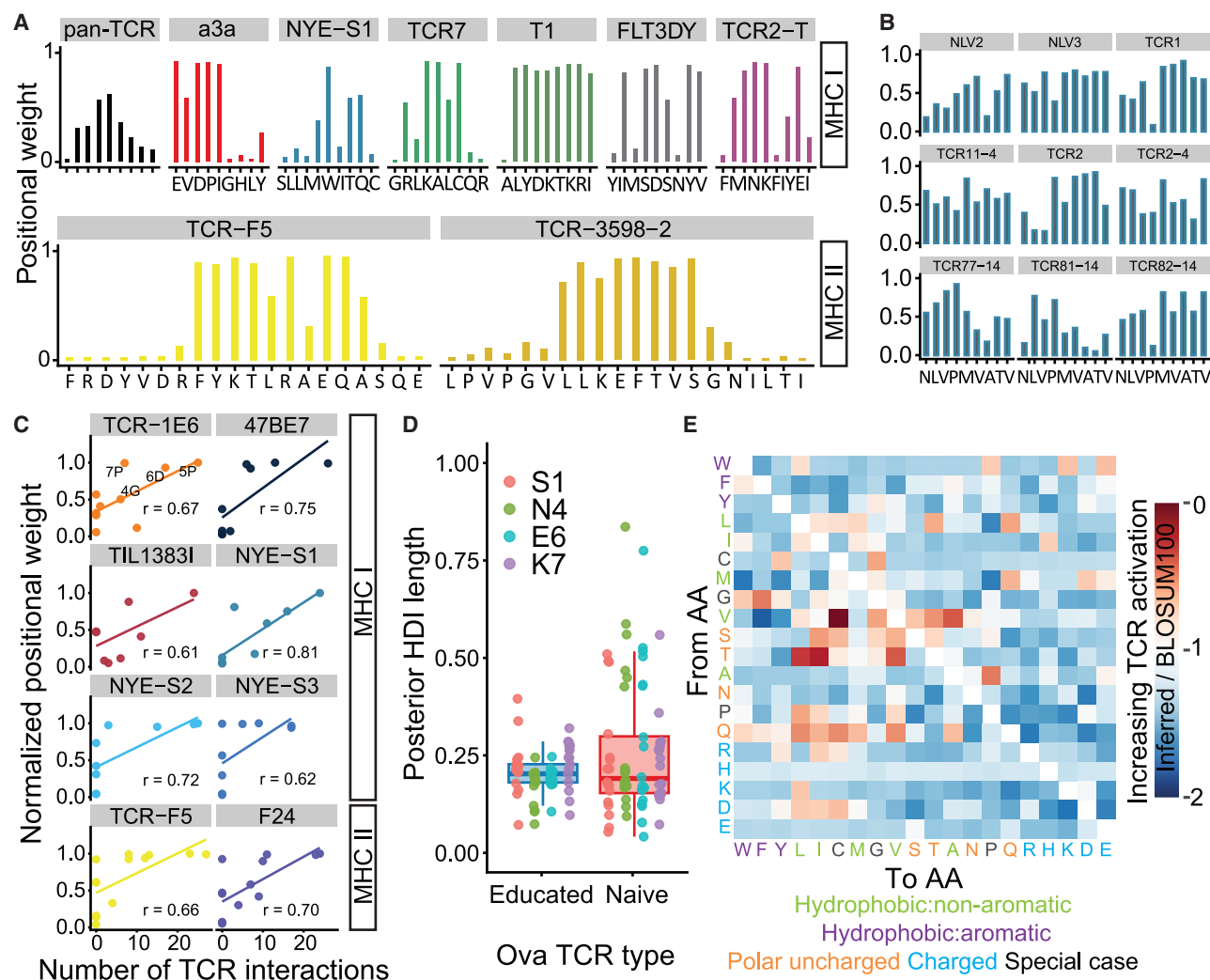


Figure 3. Inferred Batman parameters capture interpretable biochemical features of TCR-pMHC interactions

(A) Pan-TCR positional weights inferred from all 9-mer-binding TCRs in benchmark dataset (first panel), and examples of TCR-specific positional weights of selected TCRs with different index peptides (remainder of panels).

(B) TCR-specific positional weights of nine different NLV-specific TCRs, showing important peptide residues for TCR binding.

(C) Inferred positional weights ($n = 10$ positions for TCR-1E6, $n = 13$ positions for TCR-F5 and F24, and $n = 9$ positions for each of the rest 5 TCRs) correlate the number of interactions of peptide with TCR chains (derived from TCR-pMHC structure). To make different weight profiles comparable across TCR panels, we normalized positional weights by their maximum over all positions for each weight profile. 4 TCR-binding peptide residues are highlighted for TCR-1E6.

(D) 90% high-density intervals (HDIs) of positional weight posterior distributions for non-MHC-anchor positions are wider for low-avidity “naive” SIINFEKL-specific TCRs than SIINFEKL-expanded “educated” ones.

(E) Example ratio of inferred matrix elements to BLOSUM100, the best-performing conventional aa distance matrix, for within-TCR classification (linear unitless color scale).

information can be used to predict TCR activation to the many other peptides not assayed?

We developed a BATMAN-guided active learning (AL) framework (called “BATMAN-AL”), which iteratively selects a small number of mutant peptides that provide the maximum information about the cross-reactivity of a given TCR. Intuitively, these peptides should (1) be diverse, so as to adequately cover the peptide space, and (2) lie near current decision boundaries, thus helping classifiers better delineate between activation classes. Specifically, in each experimental round, the TCR binding of

a set of mutant peptides is determined experimentally, and then the binding results are fed-back to update the model, which subsequently picks the next set of maximally informative peptides to test in the next round, until predictive performance converges. Computationally, it may be advantageous to select only one peptide per round, so that the model can use binding information from this peptide to better select the next peptide; however, this may be experimentally prohibitive because of constraints on the total number of rounds a TCR can be assayed, thus requiring the selection of multiple peptides per round. To address these

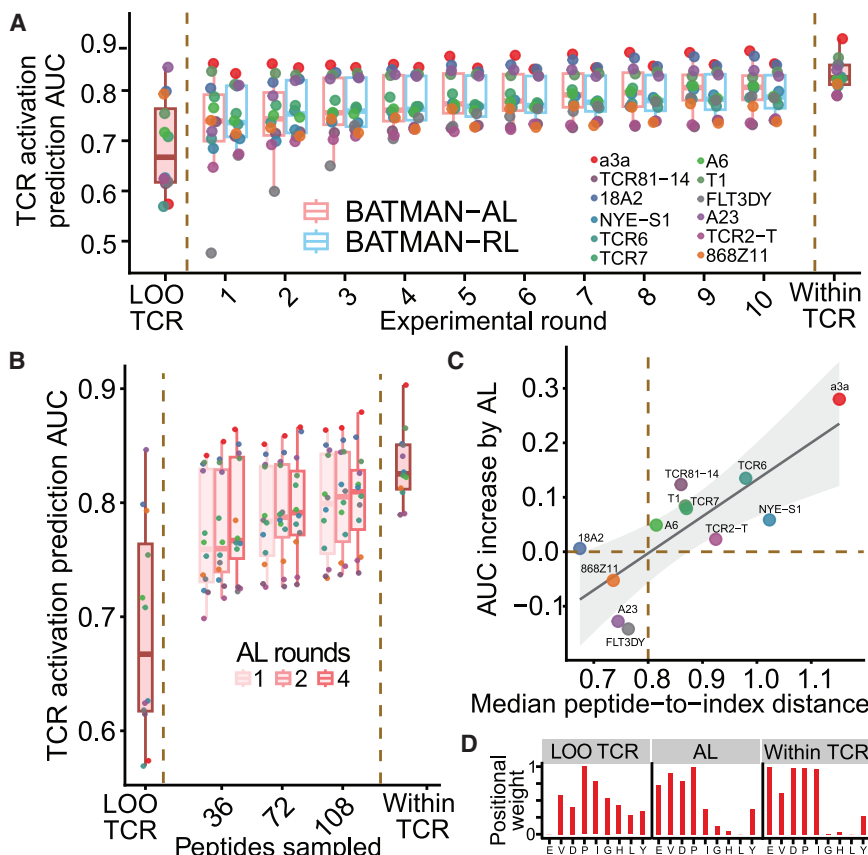


Figure 4. AL with Batman predicts TCR-specific activation with limited training samples

(A) Average AUCs for Batman with active (AL) and random learning (RL). At each learning step (experimental round, x axis), 9 peptides are sampled.

(B) BATMAN-AL performance when the same total number of peptides are sampled over different numbers of AL rounds. For performance comparison in (A) and (B), LOO-TCR and within-TCR AUCs are shown.

(C) Larger median peptide-to-index distances are associated with higher AL performance boost with Batman.

(D) After only one AL round with 9 peptides, the positional weight profile of the a3a TCR approximates the within-TCR weight profile.

0.798 for 4 rounds), although for fewer total peptides (e.g., $N_p = 36$), a single AL round is sufficient. Testing more peptides, unsurprisingly, improves performance, although if a large number of peptides can be tested (e.g., $N_p = 108$), a single AL round is again sufficient to achieve within-TCR-level performance.

What determines how much BATMAN will benefit from AL for a novel TCR? We found that larger median peptide-to-

requirements, we developed AL strategies to maximize classification performance boost of each AL round by sampling the most informative and diverse set of peptides in each round (Figure S9).

BATMAN-AL classification achieved a mean AUC of 0.786, only slightly lower than the AUC for within-TCR task (which used 144 training peptides), using only ~ 45 total peptides per TCR selected over 5 rounds (Figure 4A). Even if constrained to only select 9 total peptides in a single round, BATMAN-AL achieved a mean AUC of 0.730, outperforming the LOO-TCR mean AUC of 0.687 (Figure 4A), which suggests that even a minimal amount of AL can boost performance. BATMAN-AL also outperformed random AL (BATMAN-RL; Figure 4A), where training peptides are chosen randomly. Furthermore, BATMAN-AL demonstrated much less variance across TCRs compared with BATMAN under the LOO-TCR mode, and the variance was comparable to within-TCR levels after only 5 rounds, demonstrating how small, carefully chosen training set sizes can boost robustness across TCRs.

We next explored the performance of BATMAN-AL under various constraints on the total number of peptides to test and number of experimental rounds. For example, if a fixed number of peptides are tested (say, N_p), is it best to select all N_p of them in a single round, or might it be better to select $N_p/4$ peptides per round over 4 rounds? In general, we found that performing more rounds of experiments for a fixed total number of peptides improves predictive performance (Figure 4B; e.g., for $N_p = 72$, mean AUC increases from 0.787 in single round to

index distances of all single-aa mutants of its index peptide (calculated using the pan-TCR weight profile and aa substitution matrix) correlated with performance boost with one AL round with 9 peptides (Figure 4C). One reason for this could be that larger median distances indicate a broad range of effects for single-aa mutants (Figure S9A) and thus better separability between classes that can be learned by AL. Furthermore, TCRs with smaller median peptide-to-index distances often corresponded to narrower within-TCR-activation-class distance distributions (e.g., among strong-activating mutants of TCRs A23 and 18A2 in Figure S9A), and so already better LOO-TCR classification performance than other TCRs, gaining only a slight additional performance boost by AL (Figure 4A). For example, TCR a3a, having the largest median peptide-to-index distance, demonstrated the most improvement with AL, with its AUC approaching within-TCR levels after one AL round using 9 peptides (Figure 4A). Comparison of a3a positional weight profiles (Figure 4D) showed that after the single AL round, the weight profile indeed closely matched the within-TCR profile.

In summary, BATMAN-AL can efficiently learn the binding profile of novel TCRs from only a handful of examples and can provide custom testing strategies based on experimental constraints.

BATMAN TCR activation predictions generalize to multiple peptide mutations

While BATCAVE records effects of single-aa mutations on activation of individual TCRs, we asked whether BATMAN

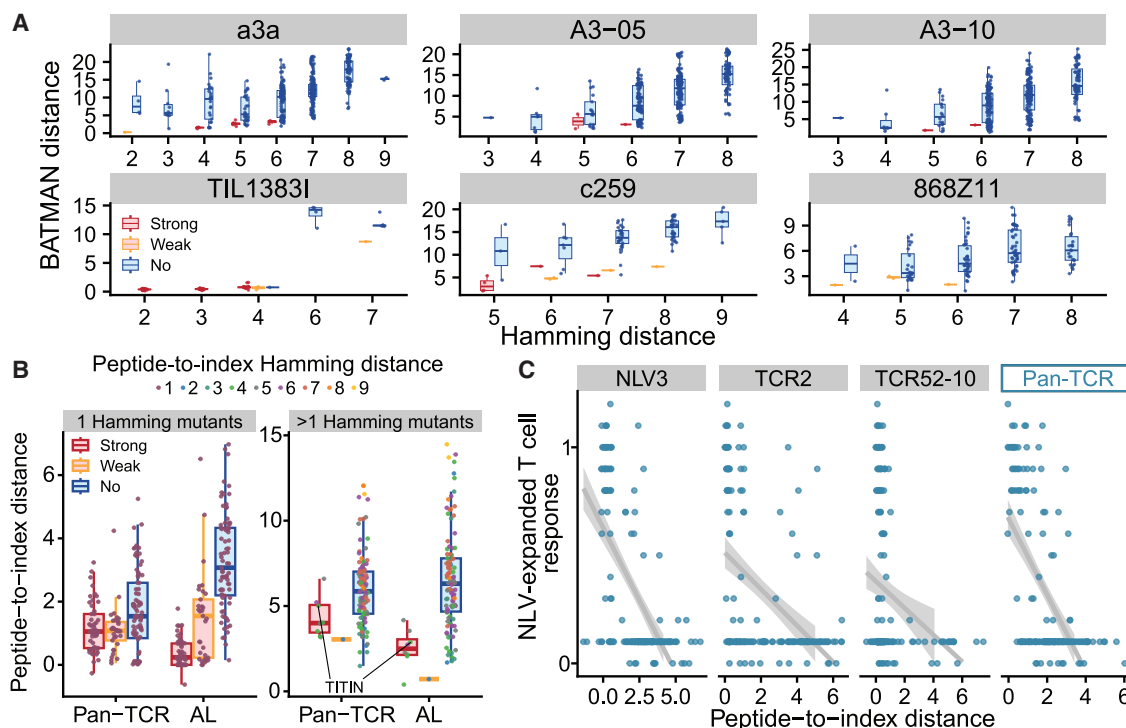


Figure 5. Batman predictions generalize to multiple-aa mutations and polyclonal T cell response

(A) Batman peptide-to-index distances from single-aa mutant training separates multi-aa mutants according to their TCR activation.

(B) BATMAN-AL with only 9 peptides for the a3a TCR allows improved TCR activation prediction of both 1 Hamming ($n = 163$) and >1 Hamming ($n = 222$) mutant peptides.

(C) NLV-expanded polyclonal T cell repertoire response against NLV mutant peptides correlates with the response of individual NLV-specific TCRs to various degrees (shown are results from 3 random example TCRs), whereas pan-TCR response, predicted from pooling across all 25 NLV-specific BATCAVE TCRs, correlating more strongly.

predictions can be generalized beyond single-aa mutants. For this benchmark, we collected multi-aa mutant TCR activation datasets for the BATCAVE TCRs for which single-aa mutants were evaluated by the same assay in the same source publication ($n = 4$ TCRs) or with the binding category marked clearly in the source publication ($n = 2$ TCRs). While BATMAN benchmarked on single-aa mutants of unique index peptides showed improvement with pooling across TCRs (Figure 2E), here, we used unpooled BATMAN because (1) the training set is biased in the number of TCRs per index peptide (3 MAGE-A3-specific TCRs, but only one specific TCR for the 3 other index peptides), and (2) the training set consists of full single-aa data while the test set consists of all available multi-aa mutant data. Thus, for each TCR, we trained BATMAN in unpooled within-TCR mode over the full single-aa data of individual TCRs and inferred the TCR-specific positional weights and aa matrix as before. We defined the peptide-to-index distance of multi-aa mutants by summing over the product of positional weight and aa substitution distance for all mutational positions. We then used the peptide-to-index distances to rank multi-aa mutants and investigated whether this ranking predicts TCR activation.

We found that BATMAN could identify strong TCR-activating mutants at all Hamming distances (Figure 5A). For example, among 32 Hamming distance 8 mutants for TCR c259 (index peptide SLLMWITQC), the sole binder NVSLWLSAV⁵² was

correctly ranked as top. As another motivating example, we considered the a3a TCR, which was designed to target the cancer-testis antigen MAGE-A3 (EVDPIGHLY). However, it resulted in fatal off-target cross-reactivity to muscle antigen Titin (ESDPIVAQY), differing from MAGE-A3 by 4 aa mutations.^{53,54} We found that BATMAN, trained on only single-aa mutants, ranked Titin 6th among 29 4-aa mutants and 8th among all 111 multi-aa mutants in our a3a dataset. These results indicate that a complete single-aa mutational scan can generalize to some extent to multi-aa mutated cross-reactive targets using BATMAN.

Can BATMAN-AL better identify TCR-activating multi-aa mutants from only single-aa mutant training? We tested this using the a3a TCR and found that the peptide-to-index distances of BATMAN-AL, using only 9 single-aa mutants, performed better than pan-TCR distances in classifying multi-aa mutant activation, with the rank of Titin improving from 41st to 16th among 111 multi-aa mutants in our a3a dataset (Figure 5B). These results suggest that BATMAN and BATMAN-AL predictions can identify cross-reactive targets for a novel TCR beyond those in the 1-Hamming space.

BATMAN predicts polyclonal T cell response

While we have thus focused on predicting activation of individual TCR clones against index peptide mutations, in some situations

(e.g., predicting neoantigen immunogenicity and viral escape mutants), we are interested in the collective response of T cell clones present in a polyclonal repertoire with different frequencies. For example, if a viral peptide elicits strong response from the peptide-specific polyclonal T cell repertoire, can BATMAN predict which mutants of that peptide could fail to elicit response from the repertoire, and thus may lead to viral escape? For the Human Cytomegalovirus peptide (NLVPMVATV), BATCAVE contained full mutational scan training data for 25 individual TCRs. For testing BATMAN predictions on polyclonal T cell response, we collected the T-Scan assay dataset measuring continuous-valued, NLV-expanded polyclonal TCR repertoire response toward all single-aa NLV mutants individually.⁵⁵

How well can individual NLV-specific TCR clone activation data in BATCAVE predict polyclonal NLV-specific TCR response? To test this, we trained BATMAN in within-TCR mode, over the full single-aa mutant data for individual TCR clones and inferred the aa matrix and TCR-specific positional weights as before. We then used the resulting weights to calculate TCR-specific peptide-to-index distances for all NLV mutants and calculated their correlations with the polyclonal T cell response. Different NLV-specific TCRs recognize different NLV mutants^{17,55} and thus are representative of the full polyclonal response to various degrees. This was reflected in the variable correlations of the peptide-to-index distances from individual TCRs with the polyclonal response (e.g., $r = 0.63$ for NLV3 but $r = 0.47$ for TCR2 and $r = 0.25$ for TCR52-10; Figure 5C), indicating that individual NLV-specific TCR clone activation data to NLV mutants cannot be reliably used to predict polyclonal peptide-specific T cell responses.

In contrast, when training BATMAN in pan-TCR mode on all NLV-specific TCR data in BATCAVE, pan-TCR predictions correlated strongly ($r = 0.61$; Figure 5C) to the polyclonal response. This indicates that pooling across TCRs is again critical, as the pan-TCR prediction found from pooling across individual TCRs statistically resembles the polyclonal response more accurately than individual TCRs. These results suggest that BATMAN provides useful predictions about polyclonal T cell response, peptide immunogenicity, and potential viral escape mutants.

DISCUSSION

We curated BATCAVE, a comprehensive database of experimentally validated positive and negative TCR-pMHC interactions from mutational scan and TCR-pMHC binding affinity assays. BATCAVE contains raw and normalized TCR activation data for both single- and multi-aa peptide mutations, and for both MHC I-restricted and MHC II-restricted TCRs. In addition, BATCAVE contains full TCR sequence information and pMHC-binding affinities when available. Using this database, we unraveled the nature of TCR cross-reactivity, including (1) the relationship between pMHC binding and TCR activation, (2) the relationship between TCR-pMHC binding and TCR activation, and (3) insights about how biochemical features (e.g., hydrophobicity) of the TCR's WT aa affects TCR binding.

Despite these advances, there are some biases in BATCAVE that require careful interpretation. First, many datasets included

used *in vitro* TCR-pMHC assays with, e.g., UV-mediated peptide exchange for MHC loading (see supplemental notes), and thus the TCR activation level by such pMHCs (Figure 1D) might be different than *in vivo* settings. Second, most BATCAVE TCRs belong to peripheral cytotoxic T cells, which survived thymic selection.⁵⁶ This thymic selection bias on TCR repertoire^{57–62} may affect, for example, the relationship between pMHC binding and TCR activation, or between TCR activation and TCR-pMHC-binding affinity. Third, recorded in BATCAVE as a single number, T cell activation for a given TCR-pMHC pair can be heterogeneous, depending on, e.g., expression of co-signaling molecules^{63,64} (e.g., CD5, CD8, and CD4). Notwithstanding these caveats, we believe this database will provide a strong benchmark for evaluating future methods and for uncovering further principles of TCR cross-reactivity.

Using this database, we then developed a computational method, called BATMAN, that predicts TCR activation of peptides based on their distances to the TCR's index peptide. BATMAN improved performance for predicting cross-reactivity across diverse TCRs and index peptides present in the BATCAVE database. Crucial toward BATMAN's performance were model-inferred positional weights and aa substitution matrices. In comparison, simple distance functions based on mismatch rules (e.g., Hamming distance) with equal weights assigned to each position proved far from sufficient to accurately characterize cross-reactivity. Furthermore, using the learned weights and substitution matrices allowed us to project cross-reactivity outside the single-aa mutational space, which we demonstrated by accurately ranking immunogenic peptides of clinically relevant TCRs. In addition, the parameters learned by BATMAN revealed variation in weight profiles for different TCRs, which was also supported by structural analyses. We then developed an active learning version of BATMAN, which helps BATMAN generalize to novel TCRs. This version provides an efficient way to sample from the prohibitively large antigenic space by iteratively selecting peptides to assay that provide the best improvement of novel TCR activation prediction accuracy. Overall, BATMAN fills a hitherto unoccupied niche of TCR-pMHC prediction methods by accurately discriminating between small differences in peptide sequences, which we show existing methods fail to predict, but which are essential for understanding neoantigen immunogenicity and off-target effects of TCR-based therapies.

While BATMAN peptide-to-index distance ranks peptides according to their predicted TCR activation, currently available datasets do not allow us to model a sharp distance cutoff to define cross-reactive peptides. With more data in the larger-than-one aa mutational scan space, it might be possible to infer decision boundaries despite the diversity of TCR-specific peptide-to-index distance distribution (e.g., seen from the median distances in Figure 4C). Furthermore, while BATCAVE single-aa mutant peptide data are relatively well balanced in TCR activation categories, non-binding peptides vastly outnumber true binders among multi-aa mutants (Figure 5A). Despite this imbalance, BATMAN demonstrates a strong positive predictive value (PPV) in the multi-aa mutant data. For example, while multi-aa mutants in the a3a TCR dataset contain 15 binders and 438 non-binders, selecting the top 10 peptides, ranked based on the BATMAN distance predicted from single-aa mutant training (Figure 5A), we obtain

PPV = 0.5, compared with a of PPV = 1 for the top 10 single-aa mutants predicted by AL distance (Figure 5B).

Finally, there are several directions of future work and corresponding applications enabled by this study. First, BATMAN could be improved by incorporating TCR sequence information into the model and by training on datasets from other types of experimental TCR cross-reactivity assays⁶⁵ (e.g., yeast display library enrichment,^{5,66} T-Scan,⁵⁵ and SABR⁶⁷), which sample more comprehensively outside the one aa mutational scan space. Second, insights from our database and method could be used to predict how TCR sequences may determine the relationship between TCR-binding affinity and cross-reactivity, and to design high-affinity TCRs with limited off-target cross-reactivity, which has numerous clinical applications.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Dr. Hannah Meyer (hmeyer@cshl.edu).

Materials availability

This study did not generate new materials.

Data and code availability

The fully curated database of TCR-pMHC interactions can be downloaded from https://github.com/meyer-lab-cshl/BATMAN-paper/tree/main/results_batman/tcr_epitope_datasets. All raw data generated for figures can be found at https://github.com/meyer-lab-cshl/BATMAN-paper/tree/main/results_batman/paper_figures under folders named by the corresponding figure panel number identifiers with filenames in the format raw_data_fig_*.csv. Custom analysis code was written in python (version $\geq 3.10.11$) or R (version $\geq 3.4.0$). All analyses, code and data to reproduce figures in this manuscript are available at: <https://github.com/meyer-lab-cshl/BATMAN-paper> (Zenodo release: <https://doi.org/10.5281/zenodo.15318036>). The python implementation of Batman ("pyBATMAN") can be installed from <https://pypi.org/project/pybatman/> and run locally. pyBATMAN installation instructions and input file specifications can be found at https://github.com/meyer-lab-cshl/BATMAN/blob/main/run_batman/pyBATMAN_Tutorial.ipynb and https://github.com/meyer-lab-cshl/BATMAN/blob/main/run_batman/BATMAN_AL.ipynb, respectively.

ACKNOWLEDGMENTS

We thank our lab members and the anonymous referees for discussion and feedback on method development and figure design, Paul G. Thomas and Zachary Sethna for discussion on TCR datasets, Anastasia Troshina and Vasilisa A. Kovaleva for designing the BATMAN logo, and all authors who shared their datasets with us. TCR-pMHC schematics in Figures 1B, 2B, and 2C and the graphical abstract were created with BioRender (<https://www.biorender.com/>). Funding: the research was supported by the Simons Center for Quantitative Biology at Cold Spring Harbor Laboratory; US National Institutes of Health grants S10OD028632 and 1R01AI167862; and the Simons Pivot Fellowship. This work was discussed in part at the Aspen Center for Physics, which is supported by National Science Foundation grant PHY-2210452. The funders had no role in the template design or decision to publish.

AUTHOR CONTRIBUTIONS

A.B., D.J.P., S.N., and H.V.M. conceptualized the work; A.B. developed the software; A.B. and D.J.P. designed the model; A.B. and C.L.W. implemented

the user interface with help from S.R.C.; A.B. curated the database with help from P.B.; A.B. conducted formal analyses with help from A.A. and H.V.M.; A.B., S.N., and H.V.M. wrote the original draft; all authors reviewed and edited the final draft; and S.N. and H.V.M. supervised the work.

DECLARATION OF INTERESTS

The authors declare no competing interests.

STAR★METHODS

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

- KEY RESOURCES TABLE
- METHOD DETAILS
 - Training data statistics of TCR-pMHC methods
 - TCR activation dataset collection
 - TCR activation dataset processing
 - Web application for visualizing TCR-pMHC interactions from our database
 - Amino acid hydrophobicity calculation
 - MHC binding predictions
 - Implementation and usage of pre-trained pMHC-TCR methods
 - Training and validation of BATMAN
 - Implementation and training of *pTEAM*
 - Assessing the relationship between inferred peptide weights with TCR-pMHC structure
- ADDITIONAL INFORMATION

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.cels.2025.101345>.

Received: July 2, 2024

Revised: February 21, 2025

Accepted: June 26, 2025

Published: July 25, 2025

REFERENCES

1. Sewell, A.K. (2012). Why must T cells be cross-reactive? *Nat. Rev. Immunol.* 12, 669–677. <https://doi.org/10.1038/nri3279>.
2. Wooldridge, L., Ekeruche-Makinde, J., Van Den Berg, H.A., Skowera, A., Miles, J.J., Tan, M.P., Dolton, G., Clement, M., Llewellyn-Lacey, S., Price, D.A., et al. (2012). A single autoimmune T cell receptor recognizes more than a million different peptides. *J. Biol. Chem.* 287, 1168–1177. <https://doi.org/10.1074/jbc.M111.289488>.
3. Keeton, R., Tincho, M.B., Ngomti, A., Baguma, R., Benede, N., Suzuki, A., Khan, K., Cele, S., Bernstein, M., Karim, F., et al. (2022). T cell responses to SARS-CoV-2 spike cross-recognize Omicron. *Nature* 603, 488–492. <https://doi.org/10.1038/s41586-022-04460-3>.
4. Łuksza, M., Sethna, Z.M., Rojas, L.A., Lihm, J., Bravi, B., Elhanati, Y., Soares, K., Amisaki, M., Dobrin, A., Hoyos, D., et al. (2022). Neoantigen quality predicts immunoediting in survivors of pancreatic cancer. *Nature* 606, 389–395. <https://doi.org/10.1038/s41586-022-04735-9>.
5. Yang, X., Garner, L.I., Zvyagin, I.V., Paley, M.A., Komech, E.A., Jude, K. M., Zhao, X., Fernandes, R.A., Hassman, L.M., Paley, G.L., et al. (2022). Autoimmunity-associated T cell receptors recognize HLA-B*27-bound peptides. *Nature* 612, 771–777. <https://doi.org/10.1038/s41586-022-05501-7>.
6. Ishii, K., Davies, J.S., Sinkoe, A.L., Nguyen, K.A., Norberg, S.M., McIntosh, C.P., Kadakia, T., Serna, C., Rae, Z., Kelly, M.C., et al. (2023). Multi-tiered approach to detect autoimmune cross-reactivity of

- therapeutic T cell receptors. *Sci. Adv.* 9, eadg9845. <https://doi.org/10.1126/sciadv.adg9845>.
7. Abdelfattah, N.S., Kula, T., and Elledge, S.J. (2024). T-Switch: A specificity-based engineering platform for developing safe and effective T cell therapeutics. *Immunity* 57, 2945–2958.e5. <https://doi.org/10.1016/j.immuni.2024.11.009>.
8. Hudson, D., Fernandes, R.A., Basham, M., Ogg, G., and Koohy, H. (2023). Can we predict T cell specificity with digital biology and machine learning? *Nat. Rev. Immunol.* 23, 511–521. <https://doi.org/10.1038/s41577-023-00835-3>.
9. Moris, P., De Pauw, J., Postovskaya, A., Gielis, S., De Neuter, N., Bittremieux, W., Ogunjimi, B., Laukens, K., and Meysman, P. (2021). Current challenges for unseen-epitope TCR interaction prediction and a new perspective derived from image classification. *Brief. Bioinform.* 22, bbaa318. <https://doi.org/10.1093/bib/bbaa318>.
10. Mason, D.M., and Reddy, S.T. (2024). Predicting adaptive immune receptor specificities by machine learning is a data generation problem. *Cell Syst.* 15, 1190–1197. <https://doi.org/10.1016/j.cels.2024.11.008>.
11. Drost, F., Chernysheva, A., Albahah, M., Kocher, K., Schober, K., and Schubert, B. (2024). Benchmarking of T-Cell Receptor-Epitope Predictors with ePytope-TCR. Preprint at bioRxiv. <https://doi.org/10.1101/2024.11.06.622261>.
12. Dens, C., Laukens, K., Bittremieux, W., and Meysman, P. (2023). The pitfalls of negative data bias for the T-cell epitope specificity challenge. *Nat. Mach. Intell.* 5, 1060–1062. <https://doi.org/10.1038/s42256-023-00727-0>.
13. McMaster, B., Thorpe, C., Ogg, G., Deane, C.M., and Koohy, H. (2024). Can alphaFold's breakthrough in protein structure help decode the fundamental principles of adaptive cellular immunity? *Nat. Methods* 21, 1–2. <https://doi.org/10.1038/s41592-023-02158-6>.
14. Koohy, H., Hudson, D., Lubbock, A., and Basham, M. (2023). A comparison of clustering models for inference of T cell receptor antigen specificity. Preprint at bioRxiv. <https://doi.org/10.1101/2023.08.04.551940>.
15. Grazioli, F., Mösch, A., Machart, P., Li, K., Alqassem, I., O'Donnell, T.J., and Min, M.R. (2022). On TCR binding predictors failing to generalize to unseen peptides. *Front. Immunol.* 13, 1014256. <https://doi.org/10.3389/fimmu.2022.1014256>.
16. Deng, L., Ly, C., Abdollahi, S., Zhao, Y., Prinz, I., and Bonn, S. (2023). Performance comparison of TCR-pMHC prediction tools reveals a strong data dependency. *Front. Immunol.* 14, 1128326. <https://doi.org/10.3389/fimmu.2023.1128326>.
17. Dorigatti, E., Drost, F., Straub, A., Hilgendorf, P., Wagner, K.I., Bischl, B., Busch, D., Schober, K., and Schubert, B. (2023). Predicting T cell receptor functionality against mutant epitopes. Preprint at bioRxiv. <https://doi.org/10.1101/2023.05.10.540189>.
18. Straub, A., Grassmann, S., Jarosch, S., Richter, L., Hilgendorf, P., Hammel, M., Wagner, K.I., Buchholz, V.R., Schober, K., and Busch, D. H. (2023). Recruitment of epitope-specific T cell clones with a low-avidity threshold supports efficacy against mutational escape upon re-infection. *Immunity* 56, 1269–1284.e6. <https://doi.org/10.1016/j.immuni.2023.04.010>.
19. Bentzen, A.K., Such, L., Jensen, K.K., Marquard, A.M., Jessen, L.E., Miller, N.J., Church, C.D., Lyngaa, R., Koelle, D.M., Becker, J.C., et al. (2018). T cell receptor fingerprinting enables in-depth characterization of the interactions governing recognition of peptide-MHC complexes. *Nat. Biotechnol.* 36, 1191–1196. <https://doi.org/10.1038/nbt.4303>.
20. Coles, C.H., Mulvaney, R.M., Malla, S., Walker, A., Smith, K.J., Lloyd, A., Lowe, K.L., McCully, M.L., Martinez Hague, R., Aleksic, M., et al. (2020). TCRs with distinct specificity profiles use different binding modes to engage an identical peptide-HLA complex. *J. Immunol.* 204, 1943–1953. <https://doi.org/10.4049/jimmunol.1900915>.
21. Reynisson, B., Alvarez, B., Paul, S., Peters, B., and Nielsen, M. (2020). NetMHCpan-4.1 and netMHCIIpan-4.0: improved predictions of mhc antigen presentation by concurrent motif deconvolution and integration of mhc eluted ligand data. *Nucleic Acids Res.* 48, W449–W454. <https://doi.org/10.1093/nar/gkaa379>.
22. Lever, M., Maini, P.K., Van Der Merwe, P.A., and Dushek, O. (2014). Phenotypic models of T cell activation. *Nat. Rev. Immunol.* 14, 619–629. <https://doi.org/10.1038/nri3728>.
23. Mariuzza, R.A., Agnihotri, P., and Orban, J. (2020). The structural basis of T-cell receptor (TCR) activation: An enduring enigma. *J. Biol. Chem.* 295, 914–925. <https://doi.org/10.1074/jbc.REV119.009411>.
24. Sušac, L., Vuong, M.T., Thomas, C., von Bülow, S., O'Brien-Ball, C., Santos, A.M., Fernandes, R.A., Hummer, G., Tampé, R., and Davis, S. J. (2022). Structure of a fully assembled tumor-specific T cell receptor ligated by pMHC. *Cell* 185, 3201–3213.e19. <https://doi.org/10.1016/j.cell.2022.07.010>.
25. Sibener, L.V., Fernandes, R.A., Kolawole, E.M., Carbone, C.B., Liu, F., McAfee, D., Birnbaum, M.E., Yang, X., Su, L.F., Yu, W., et al. (2018). Isolation of a structural mechanism for uncoupling T cell receptor signaling from peptide-MHC binding. *Cell* 174, 672–687.e27. <https://doi.org/10.1016/j.cell.2018.06.017>.
26. Choi, H.K., Cong, P., Ge, C., Natarajan, A., Liu, B., Zhang, Y., Li, K., Rushdi, M.N., Chen, W., Lou, J., et al. (2023). Catch bond models may explain how force amplifies TCR signaling and antigen discrimination. *Nat. Commun.* 14, 2616. <https://doi.org/10.1038/s41467-023-38267-1>.
27. Davis, S.J., and van der Merwe, P.A. (2003). TCR triggering: co-receptor-dependent or-independent?. author reply 626. *Trends Immunol.* 24, 624–626. <https://doi.org/10.1016/j.it.2003.10.009>.
28. Mørch, A.M., Bálint, S., Santos, A.M., Davis, S.J., and Dustin, M.L. (2020). Coreceptors and TCR signaling—the strong and the weak of it. *Front. Cell Dev. Biol.* 8, 597627. <https://doi.org/10.3389/fcell.2020.597627>.
29. Laugel, B., van den Berg, H.A., Gostick, E., Cole, D.K., Wooldridge, L., Boulter, J., Milicic, A., Price, D.A., and Sewell, A.K. (2007). Different T cell receptor affinity thresholds and CD8 coreceptor dependence govern cytotoxic T lymphocyte activation and tetramer binding properties. *J. Biol. Chem.* 282, 23799–23810. <https://doi.org/10.1074/jbc.M700976200>.
30. Moritz, A., Anjanappa, R., Wagner, C., Bunk, S., Hofmann, M., Pszolla, G., Saikia, A., Garcia-Alai, M., Meijers, R., Rammensee, H.G., et al. (2019). High-throughput peptide-MHC complex generation and kinetic screenings of tcra with peptide-receptive HLA-A* 02: 01 molecules. *Sci. Immunol.* 4, eaav0860. <https://doi.org/10.1126/sciimmunol.aav0860>.
31. Duan, F., Duitama, J., Al Seesi, S., Ayres, C.M., Corcelli, S.A., Pawashe, A.P., Blanchard, T., McMahon, D., Sidney, J., Sette, A., et al. (2014). Genomic and bioinformatic profiling of mutational neoepitopes reveals new rules to predict anticancer immunogenicity. *J. Exp. Med.* 211, 221–2248. <https://doi.org/10.1084/jem.20141308>.
32. Schmidt, J., Smith, A.R., Magnin, M., Racle, J., Devlin, J.R., Bobisse, S., Cesbron, J., Bonnet, V., Carmona, S.J., Huber, F., et al. (2021). Prediction of neo-epitope immunogenicity reveals TCR recognition determinants and provides insight into immunoeediting. *Cell Rep. Med.* 2, 100194. <https://doi.org/10.1016/j.xcrm.2021.100194>.
33. Chowell, D., Krishna, S., Becker, P.D., Cocita, C., Shu, J., Tan, X., Greenberg, P.D., Klavinskis, L.S., Blattman, J.N., and Anderson, K.S. (2015). TCR contact residue hydrophobicity is a hallmark of immunogenic CD8+ t cell epitopes. *Proc. Natl. Acad. Sci. USA* 112, E1754–E1762. <https://doi.org/10.1073/pnas.1500973112>.
34. Wimley, W.C., and White, S.H. (1996). Experimentally determined hydrophobicity scale for proteins at membrane interfaces. *Nat. Struct. Biol.* 3, 842–848. <https://doi.org/10.1038/nsb1096-842>.
35. Karnaukhov, V.K., Shcherbinin, D.S., Chugunov, A.O., Chudakov, D.M., Efremov, R.G., Zvyagin, I.V., and Shugay, M. (2024). Structure-based prediction of T cell receptor recognition of unseen epitopes using TCRnet. *Nat. Comput. Sci.* 4, 510–521. <https://doi.org/10.1038/s43588-024-00653-0>.
36. Dai, S., Heube, E.S., Rubtsova, K., Scott-Browne, J., Crawford, F., Macdonald, W.A., Marrack, P., and Kappler, J.W. (2008). Crossreactive t cells spotlight the germline rules for $\alpha\beta$ T cell-receptor interactions

- p>with MHC molecules.
- Immunity*
- 28, 324–334.
- <https://doi.org/10.1016/j.immuni.2008.01.008>
- .
37. Bagaev, D.V., Vroomans, R.M.A., Samir, J., Stervbo, U., Rius, C., Dolton, G., Greenshields-Watson, A., Attaf, M., Egorov, E.S., Zvyagin, I.V., et al. (2020). VDJdb in 2019: database extension, new analysis infrastructure and a T-cell receptor motif compendium. *Nucleic Acids Res.* 48, D1057–D1062. <https://doi.org/10.1093/nar/gkz874>.
38. Tickotsky, N., Sagiv, T., Prilusky, J., Shifrut, E., and Friedman, N. (2017). McPAS-TCR: a manually curated catalogue of pathology-associated T cell receptor sequences. *Bioinformatics* 33, 2924–2929. <https://doi.org/10.1093/bioinformatics/btx286>.
39. 10x Genomics (2019). LIT047_a New Way of Exploring Immunity – Linking Highly Multiplexed Antigen Recognition to Immune Repertoire and Phenotype. <https://www.10xgenomics.com/solutions/vdj/>.
40. O'Donnell, T.J., Kanduri, C., Isacchini, G., Limenitakis, J.P., Brachman, R.A., Alvarez, R.A., Haff, I.H., Sandve, G.K., and Greiff, V. (2024). Reading the repertoire: Progress in adaptive immune receptor analysis using machine learning. *Cell Syst.* 15, 1168–1189. <https://doi.org/10.1016/j.cels.2024.11.006>.
41. Calis, J.J.A., Maybeno, M., Greenbaum, J.A., Weiskopf, D., De Silva, A. D., Sette, A., Keşmir, C., and Peters, B. (2013). Properties of MHC class I presented peptides that enhance immunogenicity. *PLoS Comp. Biol.* 9, e1003266. <https://doi.org/10.1371/journal.pcbi.1003266>.
42. Weber, A., Born, J., and Rodríguez Martínez, M. (2021). TITAN: T-cell receptor specificity prediction with bimodal attention networks. *Bioinformatics* 37, i237–i244. <https://doi.org/10.1093/bioinformatics/btab294>.
43. Frankild, S., De Boer, R.J., Lund, O., Nielsen, M., and Kesmir, C. (2008). Amino acid similarity accounts for T cell cross-reactivity and for “holes” in the T cell repertoire. *PLoS One* 3, e1831. <https://doi.org/10.1371/journal.pone.0001831>.
44. Xia, H., McMichael, J., Becker-Hapak, M., Onyeador, O.C., Buchli, R., McClain, E., Pence, P., Supabphol, S., Richters, M.M., Basu, A., et al. (2023). Computational prediction of MHC anchor locations guides neo-antigen identification and prioritization. *Sci. Immunol.* 8, eabg2200. <https://doi.org/10.1126/sciimmunol.abg2200>.
45. Birnbaum, M.E., Mendoza, J.L., Sethi, D.K., Dong, S., Glanville, J., Dobbins, J., Özkan, E., Davis, M.M., Wucherpfennig, K.W., and Garcia, K.C. (2014). Deconstructing the peptide-MHC specificity of T cell recognition. *Cell* 157, 1073–1087. <https://doi.org/10.1016/j.cell.2014.03.047>.
46. Adams, J.J., Narayanan, S., Birnbaum, M.E., Sidhu, S.S., Blevins, S.J., Gee, M.H., Sibener, L.V., Baker, B.M., Kranz, D.M., and Garcia, K.C. (2016). Structural interplay between germline interactions and adaptive recognition determines the bandwidth of TCR-peptide-MHC cross-reactivity. *Nat. Immunol.* 17, 87–94. <https://doi.org/10.1038/ni.3310>.
47. Dezfulian, M.H., Kula, T., Pranzatelli, T., Kamitaki, N., Meng, Q., Khatri, B., Perez, P., Xu, Q., Chang, A., Kohlgruber, A.C., et al. (2023). TScan-II: A genome-scale platform for the de novo identification of CD4+ T cell epitopes. *Cell* 186, 5569–5586.e21. <https://doi.org/10.1016/j.cell.2023.10.024>.
48. Ranasinghe, S., Lamothe, P.A., Soghoian, D.Z., Kazer, S.W., Cole, M.B., Shalek, A.K., Yosef, N., Jones, R.B., Donaghey, F., Nwonu, C., et al. (2016). Antiviral CD8+ T cells restricted by human leukocyte antigen class II exist during natural HIV infection and exhibit clonal expansion. *Immunity* 45, 917–930. <https://doi.org/10.1016/j.immuni.2016.09.015>.
49. Bulek, A.M., Cole, D.K., Skowera, A., Dolton, G., Gras, S., Madura, F., Fuller, A., Miles, J.J., Gostick, E., Price, D.A., et al. (2012). Structural basis for the killing of human beta cells by CD8+ t cells in type 1 diabetes. *Nat. Immunol.* 13, 283–289. <https://doi.org/10.1038/ni.2206>.
50. Szeto, C., Zareie, P., Wirasinha, R.C., Zhang, J.B., Nguyen, A.T., Riboldi-Tunnicliffe, A., La Gruta, N.L., Gras, S., and Daley, S.R. (2022). Covalent TCR-peptide-MHC interactions induce T cell activation and redirect T cell fate in the thymus. *Nat. Commun.* 13, 4951. <https://doi.org/10.1038/s41467-022-32692-4>.
51. Sachs, A., Moore, E., Kosaloglu-Yalcin, Z., Peters, B., Sidney, J., Rosenberg, S.A., Robbins, P.F., and Sette, A. (2020). Impact of cysteine residues on MHC binding predictions and recognition by tumor-reactive T cells. *J. Immunol.* 205, 539–549. <https://doi.org/10.4049/jimmunol.1901173>.
52. Karapetyan, A.R., Chaipan, C., Winkelbach, K., Wimberger, S., Jeong, J. S., Joshi, B., Stein, R.B., Underwood, D., Castle, J.C., van Dijk, M., et al. (2019). TCR fingerprinting and off-target peptide identification. *Front. Immunol.* 10, 2501. <https://doi.org/10.3389/fimmu.2019.02501>.
53. Linette, G.P., Stadtmayer, E.A., Maus, M.V., Rapoport, A.P., Levine, B.L., Emery, L., Litzky, L., Bagg, A., Carreno, B.M., Cimino, P.J., et al. (2013). Cardiovascular toxicity and titin cross-reactivity of affinity-enhanced T cells in myeloma and melanoma. *Blood* 122, 863–871. <https://doi.org/10.1182/blood-2013-03-490565>.
54. Cameron, B.J., Gerry, A.B., Dukes, J., Harper, J.V., Kannan, V., Bianchi, F.C., Grand, F., Brewer, J.E., Gupta, M., Plesa, G., et al. (2013). Identification of a titin-derived HLA-A1-presented peptide as a cross-reactive target for engineered MAGE A3-directed T cells. *Sci. Transl. Med.* 5, 197ra103. <https://doi.org/10.1126/scitranslmed.3006034>.
55. Kula, T., Dezfulian, M.H., Wang, C.I., Abdelfattah, N.S., Hartman, Z.C., Wucherpfennig, K.W., Lyster, H.K., and Elledge, S.J. (2019). T-scan: a genome-wide method for the systematic discovery of T cell epitopes. *Cell* 178, 1016–1028.e13. <https://doi.org/10.1016/j.cell.2019.07.009>.
56. Ashby, K.M., and Hogquist, K.A. (2024). A guide to thymic selection of T cells. *Nat. Rev. Immunol.* 24, 103–117. <https://doi.org/10.1038/s41577-023-00911-8>.
57. Huseby, E.S., White, J., Crawford, F., Vass, T., Becker, D., Pinilla, C., Marrack, P., and Kappler, J.W. (2005). How the T cell repertoire becomes peptide and mhc specific. *Cell* 122, 247–260. <https://doi.org/10.1016/j.cell.2005.05.013>.
58. Stadinski, B.D., Shekhar, K., Gómez-Touriño, I., Jung, J., Sasaki, K., Sewell, A.K., Peakman, M., Chakraborty, A.K., and Huseby, E.S. (2016). Hydrophobic CDR3 residues promote the development of self-reactive T cells. *Nat. Immunol.* 17, 946–955. <https://doi.org/10.1038/ni.3491>.
59. Textor, J., Buytenhuijs, F., Rogers, D., Gauthier, É.M., Sultan, S., Wortel, I.M.N., Kalies, K., Fähnrich, A., Pagel, R., Melichar, H.J., et al. (2023). Machine learning analysis of the T cell receptor repertoire identifies sequence features of self-reactivity. *Cell Syst.* 14, 1059–1073.e5. <https://doi.org/10.1016/j.cels.2023.11.004>.
60. Lagattuta, K.A., Kang, J.B., Nathan, A., Pauken, K.E., Jonsson, A.H., Rao, D.A., Sharpe, A.H., Ishigaki, K., and Raychaudhuri, S. (2022). Repertoire analyses reveal T cell antigen receptor sequence features that influence T cell fate. *Nat. Immunol.* 23, 446–457. <https://doi.org/10.1038/s41590-022-01129-x>.
61. Luppov, D., Vlasova, E., Chudakov, D., and Shugay, M. (2025). Comprehensive analysis of $\alpha\beta$ T-cell receptor repertoires reveals signatures of thymic selection. Preprint at bioRxiv. <https://doi.org/10.1101/2025.02.09.635277>.
62. Lu, J., Van Laethem, F., Bhattacharya, A., Craveiro, M., Saba, I., Chu, J., Love, N.C., Tikhonova, A., Radaev, S., Sun, X., et al. (2019). Molecular constraints on CDR3 for thymic selection of MHC-restricted TCRs from a random pre-selection repertoire. *Nat. Commun.* 10, 1019. <https://doi.org/10.1038/s41467-019-08906-7>.
63. Cabezas Caballero, J., Huhn, A., Kutuzov, M.A., Andre, V., Shomuradova, A., van der Merwe, P.A., and Dushek, O. (2024). Generation of T Cells with Reduced Off-Target Cross-Reactivities by Engineering Co-signalling Receptors. Preprint at bioRxiv. <https://doi.org/10.1101/2024.10.25.620274>.
64. Singhaviranon, S., Dempsey, J.P., Hagymasi, A.T., Mandoiu, I.I., and Srivastava, P.K. (2025). Low-avidity T cells drive endogenous tumor immunity in mice and humans. *Nat. Immunol.* 26, 240–251. <https://doi.org/10.1038/s41590-024-02044-z>.
65. Joglekar, A.V., and Li, G. (2021). T cell antigen discovery. *Nat. Methods* 18, 873–880. <https://doi.org/10.1038/s41592-020-0867-z>.

66. Gee, M.H., Han, A., Lofgren, S.M., Beausang, J.F., Mendoza, J.L., Birnbaum, M.E., Bethune, M.T., Fischer, S., Yang, X., Gomez-Eerland, R., et al. (2018). Antigen identification for orphan T cell receptors expressed on tumor-infiltrating lymphocytes. *Cell* 172, 549–563.e16. <https://doi.org/10.1016/j.cell.2017.11.043>.
67. Joglekar, A.V., Leonard, M.T., Jeppson, J.D., Swift, M., Li, G., Wong, S., Peng, S., Zaretsky, J.M., Heath, J.R., Ribas, A., et al. (2019). T cell antigen discovery via signaling and antigen-presenting bifunctional receptors. *Nat. Methods* 16, 191–198. <https://doi.org/10.1038/s41592-018-0304-8>.
68. Nilsson, J.B., Kaabinejadian, S., Yari, H., Kester, M.G.D., van Balen, P., Hildebrand, W.H., and Nielsen, M. (2023). Accurate prediction of HLA class II antigen presentation across all loci using tailored data acquisition and refined machine learning. *Sci. Adv.* 9, ead6367. <https://doi.org/10.1126/sciadv.adj6367>.
69. Jensen, K.K., Andreatta, M., Marcatili, P., Buus, S., Greenbaum, J.A., Yan, Z., Sette, A., Peters, B., and Nielsen, M. (2018). Improved methods for predicting peptide binding affinity to MHC class II molecules. *Immunology* 154, 394–406. <https://doi.org/10.1111/imm.12889>.
70. Gfeller, D., Schmidt, J., Croce, G., Guillaume, P., Bobisse, S., Genolet, R., Queiroz, L., Cesbron, J., Racle, J., and Harari, A. (2023). Improved predictions of antigen presentation and TCR recognition with MixMHCpred2.2 and PRIME2. 0 reveal potent SARS-CoV-2 CD8+ T-cell epitopes. *Cell Syst.* 14, 72–83.e5. <https://doi.org/10.1016/j.cels.2022.12.002>.
71. Han, Y., Yang, Y., Tian, Y., Fattah, F.J., Itzstein, M.S.v., Zhang, M., Kang, X., Yang, D.M., Liu, J., Xue, Y., et al. (2023). pan-MHC and cross-species prediction of T cell receptor-antigen binding. Preprint at bioRxiv. <https://doi.org/10.1101/2023.12.01.569599>.
72. Trolle, T., and Nielsen, M. (2014). NetTepi: an integrated method for the prediction of T cell epitopes. *Immunogenetics* 66, 449–456. <https://doi.org/10.1007/s00251-014-0779-0>.
73. Jensen, M.F., and Nielsen, M. (2023). NetTCR 2.2-Improved TCR Specificity Predictions by Combining Pan-And Peptide-Specific Training Strategies, Loss-Scaling and Integration of Sequence Similarity. Preprint at bioRxiv. <https://doi.org/10.1101/2023.10.12.562001>.
74. Springer, I., Besser, H., Tickotsky-Moskovitz, N., Dvorkin, S., and Louzoun, Y. (2020). Prediction of specific TCR-peptide binding from large dictionaries of TCR-peptide pairs. *Front. Immunol.* 11, 1803. <https://doi.org/10.3389/fimmu.2020.01803>.
75. Pham, M.N., Nguyen, T.N., Tran, L.S., Nguyen, Q.B., Nguyen, T.H., Pham, T.M.Q., Nguyen, H.N., Giang, H., Phan, M.D., and Nguyen, V. (2023). epiTCR: a highly sensitive predictor for TCR-peptide binding. *Bioinformatics* 39, btad284. <https://doi.org/10.1093/bioinformatics/btad284>.
76. Zhang, P., Bang, S., Cai, M., and Lee, H. (2023). Context-Aware Amino Acid Embedding Advances Analysis of TCR-Epitope Interactions. Preprint at bioRxiv. <https://doi.org/10.1101/2023.04.12.536635>.
77. Zhang, Y., Jian, X., Xu, L., Zhao, J., Lu, M., Lin, Y., and Xie, L. (2023). iTcep: a deep learning framework for identification of T cell epitopes by harnessing fusion features. *Front. Genet.* 14, 1141535. <https://doi.org/10.3389/fgene.2023.1141535>.
78. Xu, Y., Qian, X., Tong, Y., Li, F., Wang, K., Zhang, X., Liu, T., and Wang, J. (2022). AttnTAP: a dual-input framework incorporating the attention mechanism for accurately predicting TCR-peptide binding. *Front. Genet.* 13, 942491. <https://doi.org/10.3389/fgene.2022.942491>.
79. Cai, M., Bang, S., Zhang, P., and Lee, H. (2022). ATM-TCR: TCR-epitope binding affinity prediction using a multi-head self-attention model. *Front. Immunol.* 13, 893247. <https://doi.org/10.3389/fimmu.2022.893247>.
80. Koyama, K., Hashimoto, K., Nagao, C., and Mizuguchi, K. (2023). Attention network for predicting T-cell receptor-peptide binding can associate attention with interpretable protein structural properties. *Front. Bioinformatics* 3, 1274599. <https://doi.org/10.3389/fbinf.2023.1274599>.
81. Yu, Z., Jiang, M., and Lan, X. (2024). HeteroTCR: A heterogeneous graph neural network-based method for predicting peptide-TCR interaction. *Commun. Biol.* 7, 684. <https://doi.org/10.1038/s42003-024-06380-6>.
82. Meng, E.C., Goddard, T.D., Pettersen, E.F., Couch, G.S., Pearson, Z.J., Morris, J.H., and Ferrin, T.E. (2023). UCSF ChimeraX: Tools for structure building and analysis. *Protein Sci.* 32, e4792. <https://doi.org/10.1002/pro.4792>.
83. Lu, T., Zhang, Z., Zhu, J., Wang, Y., Jiang, P., Xiao, X., Bernatchez, C., Heymach, J.V., Gibbons, D.L., Wang, J., et al. (2021). Deep learning-based prediction of the T cell receptor-antigen binding specificity. *Nat. Mach. Intell.* 3, 864–875. <https://doi.org/10.1038/s42256-021-00383-2>.
84. Springer, I., Tickotsky, N., and Louzoun, Y. (2021). Contribution of T cell receptor alpha and beta CDR3, MHC typing, V and J genes to peptide binding prediction. *Front. Immunol.* 12, 664514. <https://doi.org/10.3389/fimmu.2021.664514>.
85. Montemurro, A., Schuster, V., Povlsen, H.R., Bentzen, A.K., Jurtz, V., Chronister, W.D., Crinklaw, A., Hadrup, S.R., Winther, O., Peters, B., et al. (2021). NetTCR-2.0 enables accurate prediction of TCR-peptide binding by using paired TCR α and β sequence data. *Commun. Biol.* 4, 1060. <https://doi.org/10.1038/s42003-021-02610-3>.
86. Wu, K., Yost, K.E., Daniel, B., Belk, J.A., Xia, Y., Egawa, T., Satpathy, A., Chang, H.Y., and Zou, J. (2021). TCR-BERT: learning the grammar of T-cell receptors for flexible antigen-binding analyses. Preprint at bioRxiv. <https://doi.org/10.1101/2021.11.18.469186>.
87. Dash, P., Fiore-Gartland, A.J., Hertz, T., Wang, G.C., Sharma, S., Souquette, A., Crawford, J.C., Clemens, E.B., Nguyen, T.H.O., Kedzierska, K., et al. (2017). Quantifiable predictive features define epitope-specific T cell receptor repertoires. *Nature* 547, 89–93. <https://doi.org/10.1038/nature22383>.
88. Li, G., Iyer, B., Prasath, V.B.S., Ni, Y., and Salomonis, N. (2021). DeepImmuno: deep learning-empowered prediction and generation of immunogenic peptides for T-cell immunity. *Brief. Bioinform.* 22, bbab160. <https://doi.org/10.1093/bib/bbab160>.
89. Croce, G., Bobisse, S., Moreno, D.L., Schmidt, J., Guillaume, P., Harari, A., and Gfeller, D. (2023). Deep Learning Predictions of TCR-Epitope Interactions Reveal Epitope-Specific Chains in Dual Alpha T Cells. *Nat. Commun.* 15, 3211.
90. Gao, Y., Gao, Y., Fan, Y., Zhu, C., Wei, Z., Zhou, C., Chuai, G., Chen, Q., Zhang, H., and Liu, Q. (2023). Pan-peptide meta learning for T-cell receptor-antigen binding recognition. *Nat. Mach. Intell.* 5, 236–249. <https://doi.org/10.1038/s42256-023-00619-3>.
91. Albert, B.A., Yang, Y., Shao, X.M., Singh, D., Smit, K.N., Anagnostou, V., and Karchin, R. (2023). Deep neural networks predict class I major histocompatibility complex epitope presentation and transfer learn neoepitope immunogenicity. *Nat. Mach. Intell.* 5, 861–872. <https://doi.org/10.1038/s42256-023-00694-6>.
92. Glanville, J., Huang, H., Nau, A., Hatton, O., Wagar, L.E., Rubelt, F., Ji, X., Han, A., Krams, S.M., Pettus, C., et al. (2017). Identifying specificity groups in the T cell receptor repertoire. *Nature* 547, 94–98. <https://doi.org/10.1038/nature22976>.
93. Ogishi, M., and Yotsuyanagi, H. (2019). Quantitative prediction of the landscape of T cell epitope immunogenicity in sequence space. *Front. Immunol.* 10, 827. <https://doi.org/10.3389/fimmu.2019.00827>.
94. Zhang, W., Hawkins, P.G., He, J., Gupta, N.T., Liu, J., Choonoo, G., Jeong, S.W., Chen, C.R., Dhanik, A., Dillon, M., et al. (2021). A framework for highly multiplexed dextramer mapping and prediction of T cell receptor sequences to antigen specificity. *Sci. Adv.* 7, eabf5835. <https://doi.org/10.1126/sciadv.abf5835>.
95. Sidhom, J.W., Larman, H.B., Pardoll, D.M., and Baras, A.S. (2021). DeepTCR is a deep learning framework for revealing sequence concepts within T-cell repertoires. *Nat. Commun.* 12, 1605. <https://doi.org/10.1038/s41467-021-21879-w>.
96. Jiang, Y., Huo, M., and Cheng Li, S. (2023). TEINet: a deep learning framework for prediction of TCR-epitope binding specificity. *Brief. Bioinform.* 24, bbad086. <https://doi.org/10.1093/bib/bbad086>.

97. Ehrlich, R., Kamga, L., Gil, A., Luzuriaga, K., Selin, L.K., and Ghersi, D. (2021). SwarmTCR: a computational approach to predict the specificity of T cell receptors. *BMC Bioinformatics* 22, 422. <https://doi.org/10.1186/s12859-021-04335-w>.
98. Jokinen, E., Huhtanen, J., Mustjoki, S., Heinonen, M., and Lähdesmäki, H. (2021). Predicting recognition between T cell receptors and epitopes with TCRGP. *PLoS Comp. Biol.* 17, e1008814. <https://doi.org/10.1371/journal.pcbi.1008814>.
99. Giannakopoulou, E., Lehander, M., Virding Culleton, S., Yang, W., Li, Y., Karpanen, T., Yoshizato, T., Rustad, E.H., Nielsen, M.M., Bollineni, R.C., et al. (2023). A T cell receptor targeting a recurrent driver mutation in FLT3 mediates elimination of primary human acute myeloid leukemia in vivo. *Nat. Cancer* 4, 1474–1490. <https://doi.org/10.1038/s43018-023-00642-8>.
100. Finnigan, J.P., Newman, J.H., Patskovsky, Y., Patskovska, L., Ishizuka, A.S., Lynn, G.M., Seder, R.A., Krogsgaard, M., and Bhardwaj, N. (2024). Structural basis for self-discrimination by neoantigen-specific TCRs. *Nat. Commun.* 15, 2140. <https://doi.org/10.1038/s41467-024-46367-9>.
101. Kohlgruber, A.C., Dezfulian, M.H., Sie, B.M., Wang, C.I., Kula, T., Laserson, U., Larman, H.B., and Elledge, S.J. (2024). High-throughput discovery of MHC class I-and II-restricted T cell epitopes using synthetic cellular circuits. *Nat. Biotechnol.* 42, 1. <https://doi.org/10.1038/s41587-023-02119-6>.
102. Vazquez-Lombardi, R., Jung, J.S., Schlatter, F.S., Mei, A., Mantuano, N. R., Bieberich, F., Hong, K.L., Kucharczyk, J., Kapetanovic, E., Aznauryan, E., et al. (2022). High-throughput T cell receptor engineering by functional screening identifies candidates with enhanced potency and specificity. *Immunity* 55, 1953–1966.e10. <https://doi.org/10.1016/j.immuni.2022.09.004>.
103. Zhao, X., Kolawole, E.M., Chan, W., Feng, Y., Yang, X., Gee, M.H., Jude, K.M., Sibener, L.V., Fordyce, P.M., Germain, R.N., et al. (2022). Tuning T cell receptor sensitivity through catch bond engineering. *Science* 376, eabl5282. <https://doi.org/10.1126/science.abl5282>.
104. Bear, A.S., Nadler, R.B., O'Hara, M.H., Stanton, K.L., Xu, C., Saporito, R. J., Rech, A.J., Baroja, M.L., Blanchard, T., Elliott, M.H., et al. (2024). Natural TCRs targeting KRAS G12V display fine specificity and sensitivity to human solid tumors. *J. Clin. Invest.* 134, e175790. <https://doi.org/10.1172/JCI175790>.
105. Ali, M., Giannakopoulou, E., Li, Y., Lehander, M., Virding Culleton, S., Yang, W., Knetter, C., Odabasi, M.C., Bollineni, R.C., Yang, X., et al. (2022). T cells targeted to TdT kill leukemic lymphoblasts while sparing normal lymphocytes. *Nat. Biotechnol.* 40, 488–498. <https://doi.org/10.1038/s41587-021-01089-x>.
106. Cai, L., Caraballo Galva, L.D., Peng, Y., Luo, X., Zhu, W., Yao, Y., Ji, Y., and He, Y. (2020). Preclinical studies of the off-target reactivity of AFP158-specific TCR engineered T cells. *Front. Immunol.* 11, 607. <https://doi.org/10.3389/fimmu.2020.00607>.
107. Luo, X., Cui, H., Cai, L., Zhu, W., Yang, W.C., Patrick, M., Zhu, S., Huang, J., Yao, X., Yao, Y., et al. (2020). Selection of a clinical lead TCR targeting alpha-fetoprotein-positive liver cancer based on a balance of risk and benefit. *Front. Immunol.* 11, 623. <https://doi.org/10.3389/fimmu.2020.00623>.
108. Gejman, R.S., Jones, H.F., Klatt, M.G., Chang, A.Y., Oh, C.Y., Chandran, S.S., Korontsvit, T., Zakahleva, V., Dao, T., Klebanoff, C.A., et al. (2020). Identification of the targets of T-cell receptor therapeutic agents and cells by use of a high-throughput genetic platform. *Cancer Immunol. Res.* 8, 672–684. <https://doi.org/10.1158/2326-6066.CIR-19-0745>.
109. Aggen, D.H., Chervin, A.S., Insaidoo, F.K., Piepenbrink, K.H., Baker, B. M., and Kranz, D.M. (2011). Identification and engineering of human variable regions that allow expression of stable single-chain T cell receptors. *Protein Eng. Des. Sel.* 24, 361–372. <https://doi.org/10.1093/protein/gzq113>.
110. Foldvari, Z., Knetter, C., Yang, W., Gjerdingen, T.J., Bollineni, R.C., Tran, T.T., Lund-Johansen, F., Kolstad, A., Drousch, K., Klopfeisch, R., et al. (2023). A systematic safety pipeline for selection of T-cell receptors to enter clinical use. *npj Vaccines* 8, 126. <https://doi.org/10.1038/s41541-023-00713-y>.
111. Robbins, P.F., Li, Y.F., El-Gamil, M., Zhao, Y., Wargo, J.A., Zheng, Z., Xu, H., Morgan, R.A., Feldman, S.A., Johnson, L.A., et al. (2008). Single and dual amino acid substitutions in TCR CDRs can enhance antigen-specific T cell functions. *J. Immunol.* 180, 6116–6131. <https://doi.org/10.4049/jimmunol.180.9.6116>.
112. Rath, J.A., and Arber, C. (2020). Engineering strategies to enhance TCR-based adoptive T cell therapy. *Cells* 9, 1485. <https://doi.org/10.3390/cells9061485>.
113. Schub, A., Schuster, I.G., Hammerschmidt, W., and Moosmann, A. (2009). CMV-specific TCR-transgenic T cells for immunotherapy. *J. Immunol.* 183, 6819–6830. <https://doi.org/10.4049/jimmunol.0902233>.
114. Ogunshola, F., Anmole, G., Miller, R.L., Goering, E., Nkosi, T., Muema, D., Mann, J., Ismail, N., Chopera, D., Ndung'u, T., et al. (2018). Dual HLA B* 42 and B* 81-reactive t cell receptors recognize more diverse HIV-1 gag escape variants. *Nat. Commun.* 9, 5023. <https://doi.org/10.1038/s41467-018-07209-7>.
115. Singh, N.K., Alonso, J.A., Devlin, J.R., Keller, G.L.J., Gray, G.I., Chiranjivi, A.K., Foote, S.G., Landau, L.M., Arbujo, A.G., Weiss, L.I., et al. (2022). A class-mismatched TCR bypasses MHC restriction via an unorthodox but fully functional binding geometry. *Nat. Commun.* 13, 7189. <https://doi.org/10.1038/s41467-022-34896-0>.
116. Huseby, E.S., Crawford, F., White, J., Marrack, P., and Kappler, J.W. (2006). Interface-disrupting amino acids establish specificity between T cell receptors and complexes of major histocompatibility complex and peptide. *Nat. Immunol.* 7, 1191–1199. <https://doi.org/10.1038/n1401>.
117. Grogan, J.L., Kramer, A., Nogai, A., Dong, L., Ohde, M., Schneider-Mergener, J., and Kamradt, T. (1999). Cross-reactivity of myelin basic protein-specific T cells with multiple microbial peptides: experimental autoimmune encephalomyelitis induction in TCR transgenic mice. *J. Immunol.* 163, 3764–3770. <https://doi.org/10.4049/jimmunol.163.7.3764>.
118. Benati, D., Galperin, M., Lambotte, O., Gras, S., Lim, A., Mukhopadhyay, M., Nouël, A., Campbell, K.A., Lemerrier, B., Claireaux, M., et al. (2016). Public T cell receptors confer high-avidity CD4 responses to HIV controllers. *J. Clin. Invest.* 126, 2093–2108. <https://doi.org/10.1172/JCI83792>.
119. Poncette, L., Chen, X., Lorenz, F.K., and Blankenstein, T. (2019). Effective NY-ESO-1-specific MHC II-restricted T cell receptors from antigen-negative hosts enhance tumor regression. *J. Clin. Invest.* 129, 324–335. <https://doi.org/10.1172/JCI120391>.
120. Border, E.C., Sanderson, J.P., Weissensteiner, T., Gerry, A.B., and Pumphrey, N.J. (2019). Affinity-enhanced T-cell receptors for adoptive T-cell therapy targeting MAGE-A10: strategy for selection of an optimal candidate. *Oncoimmunology* 8, e1532759. <https://doi.org/10.1080/2162402X.2018.1532759>.
121. Chandler, D. (2005). Interfaces and the driving force of hydrophobic assembly. *Nature* 437, 640–647. <https://doi.org/10.1038/nature04162>.
122. Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D.M. (2017). Automatic differentiation variational inference. *J. Mach. Learn. Res.* 18, 1–45.
123. UTHealth (2025). DeepTR — bioinfo.uth.edu. <https://bioinfo.uth.edu/DeepTR/>.
124. Zhao, X., Xuan, F., Li, Z., Yin, X., Zeng, X., Chen, J., and Fang, C. (2025). A KIF20A-based thermosensitive hydrogel vaccine effectively potentiates immune checkpoint blockade therapy for hepatocellular carcinoma. *npj Vaccines* 10, 1. <https://doi.org/10.1038/s41541-024-01060-2>.
125. Mills, J.E.J., and Dean, P.M. (1996). Three-dimensional hydrogen-bond geometry and probability information from a crystal survey. *J. Comput. Aided Mol. Des.* 10, 607–622. <https://doi.org/10.1007/BF00134183>.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Raw and analyzed data	This paper	BATMAN paper GitHub, https://doi.org/10.5281/zenodo.15318036
TCR-pMHC structures	Protein Data Bank	PDB: 6rpb, 6rpa, 6rp9, 7na5, 3utt, 7rk7, 6cqn, 6cql
Software and algorithms		
python (version ≥ 669 3.10.11)	https://www.python.org/	https://www.python.org/
R (version $\geq 3.4.0$)	https://www.r-project.org/	https://www.r-project.org/
BATMAN software	This paper	BATMAN software GitHub, https://doi.org/10.5281/zenodo.15317456
netMHCpan4.1	Reynisson et al. ²¹	https://services.healthtech.dtu.dk/services/NetMHCpan-4.1/
netMHCpanII4.3	Nilsson et al. ⁶⁸	https://services.healthtech.dtu.dk/services/NetMHCIIpan-4.3/
The Immune Epitope Database (IEDB) Tools	Calis et al. ⁴¹ and Jensen et al. ⁶⁹	https://nextgen-tools.iedb.org/pipeline?tool=tc1 , http://tools.iedb.org/mhcii/
PRIME v2.0	Gfeller et al. ⁷⁰	http://ec2-18-188-210-66.us-east-2.compute.amazonaws.com:3000/
pMTnet Omni	Han et al. ⁷¹	https://dbai.biohpc.swmed.edu/pmtnet/analysis-omni.php
NetTepi v1.0	Trolle and Nielsen ⁷²	https://services.healthtech.dtu.dk/services/NetTepi-1.0/
NetTCR v2.2	Jensen and Nielsen ⁷³	https://services.healthtech.dtu.dk/services/NetTCR-2.2/
ImRex	Moris et al. ⁹	https://github.com/pmoris/ImRex
ERGO II	Springer et al. ⁷⁴	https://github.com/IdoSpringer/ERGO-II
epiTCR	Pham et al. ⁷⁵	https://github.com/ddiem-ri-4D/epiTCR
ITCep	Zhang et al. ^{76,77}	https://github.com/kbvstmd/ITCep/
AttnTAP	Xu et al. ⁷⁸	https://github.com/Bioinformatics7181/AttnTAP
ATM-TCR	Cai et al. ⁷⁹	https://github.com/Lee-CBG/ATM-TCR
DeepTR	bioinfo.uth.edu	https://bioinfo.uth.edu/DeepTR/Prediction.php
TCRPrediction	Koyama et al. ⁸⁰	https://github.com/kyoheikoyama/TCRPrediction
HeteroTCR	Yu et al. ⁸¹	https://github.com/yuzilan/HeteroTCR
TITAN	Weber et al. ⁴²	https://github.com/PaccMann/TITAN
UCSF ChimeraX	Meng et al. ⁸²	https://www.cgl.ucsf.edu/chimerax/

METHOD DETAILS

Training data statistics of TCR-pMHC methods

We compared our benchmarking dataset with the training datasets of existing TCR-pMHC interaction prediction methods (Figure 1A). We estimated (1) the total number of TCRs and pMHCs considered by each method, and (2) the statistics of all experimentally validated examples of TCR-pMHC interactions spanning their respective full training datasets (Figure 1A). In all cases, we disregarded any subsampling and any (usually negative) data that is artificially generated (e.g., by randomly pairing TCR and pMHC) without experimentally validating TCR-pMHC interaction. Note that the numbers below are approximate since we did not consider subsampling, or splitting data into folds during training. Thus, the number of experimentally validated TCR-pMHC interactions

broadly corresponded to aggregated numbers over all databases considered in the respective methods. Code and input training datasets to calculate training dataset statistics of different methods, if applicable, are available at: https://github.com/meyer-lab-cshl/BATMAN-paper/tree/main/results_batman/paper_figures/figure_1/1a.

BATMAN

For our curated database, non-activators (i.e., normalized TCR activation <0.1) were treated as negative instance and the rest were treated as positive instances.

pTEAM

The TCR-pMHC data is from mutational scans of SIINFEKL, a human neoantigen, and NLV-peptides in Dorigatti et al.¹⁷ We used normalized activations of 46.9, 66.09, and 40 as thresholds for defining binders as per the paper (data source: datasets S1–S4 files). Unique peptides: all 153 SIINFEKL mutants + 134 neoantigen mutants + 172 NLV-mutants. Unique TCRs: (15+21) ova-TCRs + 7 neo-antigen-TCRs + 20 NLV-TCRs.

IEDB

The source paper is Calis et al.⁴¹ where the training data is found in 2 Dataset files. Since the dataset does not use any TCR information, TCR count is recorded as 0 and the same convention is followed for all other immunogenicity models below.

PRIME 2.0

The source paper is Gfeller et al.,⁷⁰ where the training data is in Table S4. From the dataset, “random=1” datapoints were discarded as they were not experimentally validated (only about 6000 data points were experimentally validated, corresponding to “random=0”).

ImRex

The source paper is Moris et al.⁹ We recorded the training statistics based on the quote⁹ “This dataset was reduced to 19,842 unique CDR3-epitope pairs by selecting only human TCR sequences”. We discarded negative data from unspecific TCR sequences, which led to only positive examples remaining. To estimate representative TCR numbers, we downloaded an example train data from the GitHub link https://github.com/pmoris/ImRex/blob/master/models/pretrained/2020-07-30_11-30-27_trbmhci-shuffle-padded-b32-lre4-reg001/iteration_no_validation/train.csv, filtered it for y=1 (corresponding to known binders, whereas y=0 corresponds to data generated by random shuffle) and found the number of unique CDR3s and peptides.

pMTnet

The source paper is Lu et al.⁸³ We recorded the training statistics based on the quote⁸³ “We collected a total of 32,607 pairs of TCR–pMHC bindings from a series of peer-reviewed publications [...]. We created ten times more negative pairs by randomly mismatching the TCRs and pMHCs of these 32,607 pairs.” i.e. this dataset does not contain any experimentally verified negative data. We downloaded the training data from the GitHub link https://github.com/tianshilu/pMTnet/blob/master/data/training_data.csv and counted 29226 unique CDR3 sequences and 429 unique peptides.

epiTCR

Full experimentally validated training data was available for both negative and positive TCR-pMHC pairs. The source paper is Pham et al.⁷⁵ Training dataset was downloaded from the GitHub link <https://github.com/ddiem-ri-4D/epiTCR/blob/main/data/categories/full-training-with-categories.csv.zip> following which we counted the number of binding and non-binding CDR3β-pMHC pairs.

ERGO II

The source paper is Springer et al.⁸⁴ The training dataset does not contain any experimentally validated negative data. Since the model trained on the McPAS dataset performed better than the one trained on the VDJDdb dataset in both the original publication and on our own dataset, we recorded the statistics for the McPAS-trained model. Following the dataset reported in the supplementary table of the source paper,⁸⁴ we recorded the number of unique CDR3b sequences as the number of unique TCRs in the training data: 9490, and number of unique peptides=319, number of positive sample=11630.

NetTCR-2.0

The source paper is Montemurro et al.⁸⁵ We recorded the statistics for prediction mode ‘paired TCR’ which has superior performance in internal mode comparisons.⁸⁵ We recorded the training statistics based on the quote⁸⁵ “Positive data points were taken from ... 3859 unique binding pairs were identified from IEDB and 2843 from VDJDdb. These provided 4598 unique CDR3α-/β-peptide interactions with 276 different peptides specific to allele HLA-A*02:01.

Negatives were derived from 10X. Using the same restrictions as for the positives (CDR3 length between 8 and 18 aa, peptide length 9, and peptides specific for HLA-A*02:01), 627,323 unique data points with 0 UMI counts to all the tested peptides were identified. These contained 33,017 unique TCRs tested against a set of 19 different peptides. In total, 17 of these overlapped with the peptides in the positive data set.”

Thus, total number of TCRs=4598+33017=37615, with 4598 positive and 627,323 negative instances, and 276+19-17= 278 peptides.

TCR-BERT

The source preprint is Wu et al.⁸⁶ We disregarded the pre-training and TCR embedding learning data, and only recorded the numbers of LCMV peptide specific TCR sequences (so 1 unique peptide considered). We used the following quote to record training statistics⁸⁶: “Overall, this results in n=17,702 unique TRA/TRB pairs with consistent labels that we use for model training and evaluation. Among these, 13% (n=2306) are observed to have mid or high binding – we consider these “positive” examples of TRA/TRBs binding GP33.”

TCRdist

The source paper is Dash et al.⁸⁷ No negative data was considered, and the training data is summarized in Extended Data Table 1. We recorded from the training data summary that total number of TCRs = total number of positive examples = sum of number of clones for 10 peptide-specific repertoires = $117 + 305 + 324 + 642 + 87 + 158 + 291 + 76 + 275 + 61 = 2336$.

TITAN

The source paper is Weber et al.⁴² All negative data was generated by shuffling, and so were disregarded in reporting the training statistics. We used the following quote⁴² to record training statistics: "With this procedure, we build a training dataset of 46,290 examples, 50% of which are positive, encompassing 192 different epitopes."

DeepImmuno

This is a general immunogenicity prediction method, so no TCR information is involved. The source paper is Li et al.,⁸⁸ from which we use the following quote to record training data statistics: "Specifically, 8971 data instances were retained in the final dataset, among which 4059 were positive reactive instances and the remaining 4912 were negative".

MixTCRpred

All negative data was generated by shuffling, so we disregarded them in calculating training data statistics. To record training data statistics, we used the quote from the source preprint Croce et al.⁸⁹: "For further analysis, only pMHCs with ten or more binding TCRs were considered, resulting in a total of 17,715 $\alpha\beta$ TCRs interacting with 146 pMHCs (Figure 1C; Table S1)". We also downloaded the full training data from the GitHub link https://github.com/GfellerLab/MixTCRpred/blob/main/full_training_set_146pmhc.csv and confirmed that the total number of data points is equal to 17,715.

PanPep

The source paper is Gao et al.⁹⁰ Negative data was generated computationally, so we disregarded them in recorded training data statistics. For the rest, we used the following quote⁹⁰ to record training data statistics: "As a result, the base dataset contains 699 unique peptides and 29,467 unique TCRs with 32,080 related peptide-TCR binding pairs considering the cross-reactivity of peptide-TCR binding."

BigMHC

We collected statistics for the "Immunogenicity Training" mode and used the quote "BigMHC transfer learned only on the nonrandom pMHC data, of which 1,580 are positive and 5,293 are negative" from the source paper Albert et al.⁹¹ to record training data statistics.

catELMo

The source preprint is Zhang et al.⁷⁶ Negative data is generated randomly, so we disregarded them in calculating the training data statistics. For recording statistics of the positive data, we used the quote⁷⁶ "Altogether, we obtained 150,008 unique TCR-epitope pairs known to bind to each other having 140,675 unique TCRs and 982 unique epitopes."

ATM-TCR

The source paper is Cai et al.⁷⁹ Negative data is generated randomly, so we disregarded them in calculating the training data statistics. For recording statistics of the positive data, we used the quote⁷⁹ "This resulted in the primary dataset containing 128,142 unique TCR-epitope pairs, with 931 unique epitopes and 119,984 unique TCRs."

GLIPH

The source paper, Glanville et al.⁹². No negative data was considered and the training data was obtained from Table S1 as specified in the manuscript. Table S1, sheet="Curated", had 1,973 unique CDR3s and 7 unique peptides, with 2,067 CDR3-peptide pairs in total.

Repitope

The source paper is Ogishi and Yotsuyanagi.⁹³ Even though they consider both TCR and peptide sequences, they are not matched in the training data. We included the statistics from the training data reported in Supplementary Data sheet 2. We focused on the immunogenicity column only, counted positives and negatives and added them up.

TCRAI

The source paper is Zhang et al.⁹⁴ We downloaded the training data from the GitHub link <https://github.com/regeneron-mpds/TCRAI/blob/main/data/CNN-prediction-with-REGN-pilot-version-2.csv> and recorded the number of unique TCRs, peptides, and TCR-peptide pairs (positive and negative were based on the 'binds' column).

DeepTCR

The source paper is Sidhom et al.⁹⁵ DeepTCR has a (1) TCR encoding mode, trained on antigen-specific TCR sequences (only positive examples), and (2) a UMI regression mode, trained on the 10X data. For mode (1) (TCR encoding), the publication quotes⁹⁵ "we collected data for tetramer-sorted antigen-specific cells for nine murine (Db-F2, Db-M45, Db-NP, Db-PA, Db-PB1, Kb-M38, Kb-SIY, Kb-TRP2, Kb-m139) and seven human (A1-CTELKLSY, A1-VTEHDTLLY, A2-GILGFVFTL, A2-GLCTLVAML, A2-NLVPMTATV, B7-LPRRSGAAGA, B7-TPRVTGGAM) antigens where the ground truth label corresponds to a particular antigen specificity for an individual sequence". This set comprises all the positive data they considered. To estimate the number of TCR sequences used, we counted the total numbers found in the GitHub folders https://github.com/sidhomj/DeepTCR/tree/master/Data/Murine_Antigens and https://github.com/sidhomj/DeepTCR/tree/master/Data/Human_Antigens to find $(117 + 291 + 305 + 324 + 642 + 158) + (776 + 655 + 456 + 600) + (410 + 435 + 347) + 87 = 5,603$ murine and $25 + 271 + (275 + 588) + (76 + 739) + (61 + 166) + 214 + 65 = 2,480$ human TCRs and an equal number of positive TCR-pMHC binding. For mode (2) (regression), the publication quotes⁹⁵ "a second single-cell dataset published by 10x Genomics where the binding to cognate T cells of 57,229 unique α/β pairs to 44

specific pMHC multimers and 6 negative controls was characterized". This dataset was downloaded from the GitHub link https://github.com/sidhomj/DeepTCR/tree/master/Data/10x_Data and has continuous count information on 57,228*44 TCR-pMHC pairs. Out of them, we designated negative as count=0 ($n=2,562,701$) and positive as count > 0 ($n=298,699$). A total of 9 murine and 7-+44+6-5 (common) =52 human antigens were considered. In reporting the statistics of the deepTCR model, we combined the number of positives from the two modes ($5603+2480+298699=306782$). Total number of TCRs were also combined ($5603+2480+57228=65311$).

TEINet

No experimentally validated negative data was used. To record training data statistics, we used the following quote "At last, we constructed a large dataset with 44,682 pairs of TCRs and epitopes, among which 41,610 TCRs are linked to 180 epitopes" from the source paper Jiang et al.⁹⁶

SwarmTCR

No experimentally validated negative data was used. To record training data statistics, we used the following quote "In total, our default SC dataset comprised 1447 TCRs (for complete dataset counts see Additional file 1: Table 4)", from the source paper Ehrlich et al.⁹⁷ We found that Table 4 has 83 peptides.

TCRGP

No experimentally validated negative data was used. To record training data statistics, we used the following quote, "The VDJdb data and the Dash data have some overlap for TCRs specific to three epitopes: In the VDJdb data 34 (27 unique) of the 413 (242 unique) TCRs for pp65495-503, 30 (27) of 299 (152) for BMLF1280-288, and 74 (61) of 239 (138) for M158-66 can also be found from the Dash data" from the source paper Jokinen et al.⁹⁸ So the data sources are VDJdb and Dash data. We looked into the GitHub folder https://github.com/emmijokinen/TCRGP/tree/master/training_data/paper for data (files names marked with "unique") and found $138+474+120+234+286+542+526+586+1168+174=4248$ unique Dash TCRs (corresponding to "three epitopes from humans and seven epitopes from mice"⁹⁸) and $278+78+120+156+190+276+304+106+92+118+116+208+130+282+396+484+100+298+58+80+240+104+610=4,824$ unique VDJdb TCRs (23 peptides). So there was a total of $4,248+4,824-27-27-61=8,957$ unique TCRs, and $10+23-3=30$ peptides combining two datasets.

TCR activation dataset collection

We collected continuous TCR-pMHC datasets for complete single-aa mutational scans from all publications containing raw datasets ($n=21$). In the following, we include notes on acquisition and processing of TCR activation data from existing publications, grouped by source publication and/or TCR. Code used for acquisition of the data are available on the manuscript GitHub page at: https://github.com/meyer-lab-cshi/BATMAN-paper/tree/main/results_batman/tcr_epitope_datasets. Note that more details on processing single-aa mutant data for BATCAVE are in the next section.

FLT3DY TCR

We sourced the data from Giannakopoulou et al.⁹⁹ Figure 1H IFN- γ production heatmap. In the mutational scan matrix source data, the recorded IFN- γ values corresponding to the unmutated amino acids (denoting the index peptide IFN- γ value) are different for different positions on the peptide (likely because they are different replicates of the index peptide). We recorded the average IFN- γ value over all occurrences of the index peptide. The FLT3DY TCR sequence is patented.

BE7 TCR

We sourced the data from Finnigan et al.¹⁰⁰ Figure 5F IFN- γ percentage data, from the "Source Data" spreadsheet associated with the article. Mutant peptide IFN- γ measurements have two or three replicates, we take an average over all available replicates. The 47BE7 TCR sequence is recorded from Figure 2A.

a3a TCR

We sourced the mutational scan data for a3a TCR from Kohlgruber et al.¹⁰¹ Figure 6B peptide fold enrichment data, from the "Source Data" spreadsheet associated with the article. Note that we skip positions N1, N2, C1, C2 and only consider positions marked 1–9. We also minmax normalized the fold enrichment data to lie between [0,1]. The a3a TCR sequence is recorded from Figure S3(a) of Vazquez et al.¹⁰²

Additionally, we collected multi-aa mutant peptide activation data for the a3a TCR, with activation labels marked by the source publication or assigned by us as follows:

- The Tables S2 and S6 of Cameron et al.⁵⁴: we recorded the TCR activation labels as 'Yes': SB, 'Yes,low':WB, and 'X':NB.
- Figure 6D and Table S12 list of Zhao et al.¹⁰³: Out of all tested peptides, only the ones marked in color in Figure 6D are recorded as SB, the remainder are recorded as NB.
- Figure 6a of Kohlgruber et al.¹⁰¹: only the exact epitope sequences marked in color are recorded as SB. We neglected all other data in the figure, since the authors performed peptide tile scans involving 90-aa-long sequences, and the exact epitopes, if any, were not identified by them for non-enriched tiles.
- Figures 5C and 5E of Vazquez et al.¹⁰²: among the tested epitopes in Figure 5C, the ones with $p < 0.0001$ in Figure 5E are recorded as SB, the remainder as NB. Additionally, Figure 6E (and the corresponding peptides listed in Table S4), for which, according to the paper, "In addition to epitopes derived from MAGE-A6 and titin, TnT-TCRa3a cells responded to additional peptides (i.e., with higher than 10% activity relative to the MAGE-A3168–176 target), which included the recently reported off-target FAT2, as well nine additional peptides (ANR16, CD166, COG4, CSPG2, FAT1, IL7RA, PLD5, RPAB2, and RUSD2).

In contrast, only three and two off-target peptides in addition to the highly homologous MAGE-A6168–176 peptide activated TnT-TCRA3-05 and TnT-TCRA3-10 cells, respectively.” So, peptides with <10% activity are recorded as NB, while >10% is recorded as SB.

Note that the peptide EVDPIRHYY had conflicting labels among the above sources. In Zhao et al.,¹⁰³ it is negative (within DMSO control), while in Cameron et al.⁵⁴ its response is recorded “Yes, low” and it is marked as enriched in Kohlgruber et al.¹⁰¹ We record it as a WB.

APN- and EWW-TCR

We recorded the log2FC data corresponding to Figures 1B and 1C of Bentzen et al.¹⁹ We saved the heatmaps as webp files, converted them to jpg files, read pixel-wise RGB values using custom Python code and then mapped the RGB values to log2FC values using the colorbar. Since Figures 1A and 1B shows that unspecific TCRs have logFC at zero, we set all negative log2FC values as 0, and set that as minimum of logFC. We then within-TCR minmax normalized the log2FC values to lie between [0,1]. The publication notes “TCR sequences and expression vectors must be obtained through a material transfer agreement.” So we did not include the TCR sequences in our database.

KRAS TCRs

We sourced the mutational scan data from NFAT activation heatmaps of Bear et al.,¹⁰⁴ found in Figure 2C. The unnormalized raw data is found in the supplementary “View Supplementary Data values” Figure 2C sheet. Note that cysteine (C) substitution results are separately recorded in the right of the data sheet. Furthermore, for non-C substitutions, while the alphabetic aa order is maintained for the substituted amino acid, in recording the data in the 18 × 10 (as opposed to 20 × 10, or 19 × 10, excluding only C) grid for each TCR, C and the WT amino acids are missing for each position. So, the set of substituted amino acids is different for each position, and that is not accurately recorded in the grid row amino acid name, since the amino acids cannot be common for all columns. We recorded the data assuming that the amino acid alphabetic order is maintained for each column corresponding to a specific peptide position, with cysteine and the WT aa at that position missing.

Following the legend in Figure 2C, we minmax normalized the results using the controls for each TCR: Minimum (Jurkat Background) and Maximum (PMA/I avg), noted separately on the right side of the grid in the spreadsheet. The index peptide activity is also found in the same location. Some peptide activities attain a small negative value after this normalization, we set them to zero in our record. TCR sequences are found from Table 1. In Table 1, the epitope for TCR A11Vc is listed as 8-16V, while for the rest of the TCRs it is 7-16V. Since Figure 2C performs mutational scan over 7–16 for all TCRs, we record 7–16V as the index peptide for all TCRs.

T1 and T3 TCRs

We sourced the data from Ali et al.¹⁰⁵ IFN- γ production heatmaps in Figure 2G. In the source data for the mutational scan matrix, the recorded IFN- γ values corresponding to the unmutated amino acids (denoting the index peptide IFN- γ value) are different for different positions on the peptide (likely because they are different replicates of the index peptide). We recorded the average IFN- γ value over all occurrences of the index peptide. The T1 and T3 TCR sequences are patented.

TCR* TCRs

We sourced the data from Łuksza et al.⁴ Figures 3D and 3F EC50 heatmaps. We recorded the normalized $-\log_{10}$ (EC50) fitted values for mutant peptides, found in Source Data Extended Data Figure 6, where EC50 was estimated by the fit parameter named “ K_a ”. Following the description in the source text⁴ with “the inferred EC50’s were further clipped to the range of 1E-4 μ g/mL to 1E+4 μ g/mL”, we applied the same restriction. We first calculated $a = -\log_{10}$ (EC50) for all mutant peptides corresponding to a given TCR, then clipped the values (including $\pm \text{Inf}$) to $[-4, 4]$ and then mapped all recorded a values to $[0, 1]$ for a given TCR. The TCR V β and V α sequences were sourced from Extended Data Figure 4. We aligned known VDJ allele sequences to the TCR sequences to infer the alleles and separate the CDR3 sequences.

TCR*-T TCRs

We sourced the data from Cai et al.¹⁰⁶ Figures S1A and S1B IFN γ Mutant/WT ratio barplots. As the source data was not available to us, we opened the figure files in Adobe Illustrator and recorded the total height of the bars in pixels (discarding error bars) and converted them into IFN γ values, using the index peptide amino acid bars at each position as the standard.

The TCR CDR sequences were taken from a companion article, Luo et al.¹⁰⁷ Figure 1A. We designated the regions corresponding to sequences starting with “CAAS...” or “CASS...” before “G*G” as CDR3 regions and aligned known VDJ allele sequences to the rest of the TCR sequences to infer the alleles.

A6 and B7 TCRs

We sourced the raw data from Gejman et al.¹⁰⁸ Figures 3E and 3F heatmaps for minigene fold change, which was kindly provided to us by the authors. We recorded the normalized $\log_2$ of depletion values for mutant peptides. We first recorded the depletion value for all mutant peptides for a given TCR and then normalized $-\log_2$ (depletion) values to map all recorded values between $[0, 1]$ for a given TCR. For B7, we clipped the depletion values at 1.5 before normalizing, to minimize the effect of one outlying data point at 1.8 on discrete TCR activation levels.

The TCR sequences were sourced from Table S3. For A6, we recorded the TCR V β and V α sequences corresponding to the bacterial expression construct (for which, α and β chain sequences are separately given). We aligned known VDJ allele sequences to the TCR V β and V α sequences to infer the alleles and separate CDR3 sequences.

NYE-S* TCRs

We sourced the raw data from Coles et al.²⁰ Figures 4D–4F IFN γ release heatmaps, which was kindly provided to us by the authors. TCR sequences are in Table II of the paper.

pTEAM TCRs

For ovalbumin-specific ('SIINFEKL') TCRs, we sourced the raw data from Straub et al.¹⁸ Figures 5D and S5B NFAT expression heatmaps, which was kindly provided to us by the authors. The TCR sequences were sourced from another preprint, Dorigatti et al.,¹⁷ Data S1 and S2. We sourced the NLV-specific TCR data, from Figures 5A and S17A NFAT data of Dorigatti et al.¹⁷ NFAT data and TCR sequence data were sourced from https://github.com/SchubertLab/TcrPrediction_MutatedAPLs/tree/master/data. We recorded the unnormalized mutational scan data.

Z11 TCR

We sourced the TCR activation and TCR-pMHC dissociation constant data from Moritz et al.³⁰ Figure 6B sheet of the Supplementary Data Sheet. This work reports both binding constant K_d and TCR activation (NFAT-luc2 luminescence) simultaneously for all mutant peptides, of which we record both. We recorded TCR activation value at TCR concentration 100pM, which is the activation threshold for most mutant peptides based on Figure 6E of the source publication.³⁰ Note that the amino acid cysteine is missing from the mutational scan due to its tendency to dimerize.

The TCR used here is a single chain TCR, with its construct described in Material S3 of the source publication.³⁰ We referred to Aggen et al.¹⁰⁹ Figure S1 for the full TCR sequence. We aligned known VDJ allele sequences to the TCR V β and V α sequences to infer the alleles and separate CDR3 sequences. We designated the sequence "CASSDTSYEQYF" before the sequence "G*G..." as CDR3 β in V β .

Additionally, we sourced multi-aa mutant peptide TCR activation from Supplementary Data "Figure 6d" sheet is taken from Moritz et al.³⁰ Similar to single-aa mutants, we recorded TCR activation value at TCR concentration 100pM. We subsequently normalized the multi-aa mutant TCR activation values by the maximum of single-aa mutant peptide TCR activation, and assigned activation categories based on the normalized TCR activation following the same convention as single-aa mutants (Figure 1C).

E6 TCR

We sourced the data from Bulek et al.⁴⁹ Figure 2 mutational scan heatmap for TNF secretion. The TCR sequence was sourced from the Proten Data Bank entry 5c07, website: <https://www.rcsb.org/structure/5c07>.

A23 and 1G4 (c259) TCRs

We sourced the data from Foldvari et al.¹¹⁰ Figures 2A and 2D IFN- γ heatmap (presented in Tables S1 and S3). Note that some amino acids are missing in mutational scan of A23 (e.g., Y at position 1). To record IFN- γ values, we converted the original PDF file to excel spreadsheet with Adobe PDF exporter and changed the commas (",") to decimal points (".") in the data. We referred to Figure S1 of the source publication¹¹⁰ for TCR sequences. We aligned known VDJ allele sequences to the TCR V β and V α sequences to infer the alleles and separate CDR3 sequences. Note that the CDR3a region of 1G4 in supplementary reads "CAV-RPLYGGSYIP", which indicates a substitution TS $\rightarrow$ LY in the original 1G4 sequence (e.g., found in Figure 1B of Robbins et al.,¹¹¹ or PDB: 2BNU, <https://www.rcsb.org/structure/2BNU>). This variant of the TCR is known as c259,¹¹² hence we record that name here instead of 1G4.

Additionally, multi-aa mutant peptide activation data for c259 is sourced from Figure 4B, and Supplementary Data Sheet 1 of Karapetyan et al.⁵² Of these peptides, for the ones in the supplementary, the paper mentions "None of the tested peptides showed evidence of meaningful AKD10R3 cell activation, with a median activation of 1.2% across 65 tested peptides." So we designate them as NB. For the rest, the paper mentions, "out of 16 peptides with high recognition scores, 44% (7/16) demonstrated at least 75% activation (6, including the original peptide, demonstrated activation above 100%), 27% (4/16) were demonstrated activation in the range of 20–40%, and 33% (5/16) displayed either weak (<15%) activation or no activation at all (Figure 4B)." We designate these three classes as SB, WB, and NB respectively.

While the above single-aa mutation scan data corresponds to SLLMWITQC as the index peptide, for TCR c259, we found matched TCR activation and TCR-pMHC dissociation constant measurements for single-AA mutants of the peptide SLLMWITQV in Cabezas et al.,⁶³ which we recorded separately for plotting Figure 1E. For this purpose, we recorded K_d values from Table S2 (geometric mean) of the source paper.⁶³ The TCR activation data is available from Figure 5C killing assay heatmap of the source paper.⁶³ The raw data is not available, neither are the heatmaps in vector graphics format. So we saved the heatmaps as jpg files, read pixel-wise RGB values using a Python code and then map the RGB values to killing assay values using the colorbar. Cysteine is excluded in the scan at all positions.

NLV* TCRs

We sourced the data from Kula et al.⁵⁵ mutational scan heatmaps of mutant peptide enrichment in Figure S5. Note that, there were two kinds of enrichment scans done in the paper, with slightly different results: one for 56-aa long fragments (Figures 5C and 5D) and one for 9-aa long fragments (Figure S5), we recorded the latter. Corresponding TCR sequences were recorded from Schub et al.¹¹³ Table II.

***A* and *D* TCRs**

We sourced the data from Ogunshola et al.¹¹⁴ mutational scan heatmaps of NFAT luminescence in Figure 5 (raw data was found in Data S1 "TCR recognition_normalized" datasheet). TCR sequences were recorded from Figure 4A.

TIL1383I TCR

We recorded the mutational scan data for TCR TIL1383I from Figure 5A ELISA IL-2 data of Singh et al.,¹¹⁵ with the raw data kindly provided by the corresponding author. TIL1383I TCR alpha and beta sequences are recorded from PDB: 7RK7, Website: <https://www.rcsb.org/fasta/entry/7rk7/display>. We aligned known VDJ allele sequences to the TCR V β and V α sequences to infer the alleles and separate CDR3 sequences.

Additionally, the table in Figure 6 of Singh et al.¹¹⁵ contains IL-2 data for multi-aa mutant peptides, with the index peptide IL-2 normalized to 100. So, following our usual normalization scheme (Figure 1C) we record the activation categories based on IL-2 measurements as follows: <10: NB, 10–50: WB, and $\geq$ 50: SB.

Finally, we derived TCR-pMHC dissociation constants K_d for mutant peptides from the $\Delta\Delta G$ heatmap of Figure 5B. The raw data are not available, neither are the heatmaps in vector graphics format. So we saved the heatmaps as webp converted to jpeg file, read pixel-wise RGB values using a Python code and then map the RGB values to $\Delta\Delta G$ values using the colorbar. From $\Delta\Delta G$ data, we calculated K_d by using the equation $K_d = \exp(\Delta G / RT)$, with $\Delta G = \Delta G_{WT} + \Delta\Delta G$, where ΔG of WT peptide is -6.25 kcal/mol from Figure 6 table and T is assumed to be 298K, with $RT = 0.592$ kcal/mol (this also matches Figure 6 ΔG and K_d conversions).

A3-05 and A3-10 TCRs

We recorded the %CD69 data corresponding to Figure 6A in the paper Vazquez-Lombardi et al.¹⁰² The raw data is not available, neither are the heatmaps in vector graphics format. So we save the heatmaps as jpg files, read pixel-wise RGB values using a custom Python code and then map the RGB values to %CD69 values using the colorbar. Note that we skipped a3a mutational scan data, which is already recorded from Kohlgruber et al.¹⁰¹ as described above, and shows very similar results between Figure 6b of Kohlgruber et al.¹⁰¹ and the corresponding values in Figure 6a of Kohlgruber et al.¹⁰² The relevant TCR sequences are patented.

Additionally, Figure 6E (and the corresponding peptides listed in Table S4) of Vazquez et al.¹⁰² contains A3-05 and A3-10 TCR activation data for multi-aa mutant peptides. For recording the TCR activation levels of these peptides, we followed the same scheme as a3a TCR multi-aa mutant peptide activation data, sourced from the same figure and described above.

3K-specific TCRs

We sourced the mutational scan data from Figure 2 of Huseby et al.⁵⁷ The data is presented as a categorical heatmap of 3 T cell response classes. We recorded, with index peptide activity set to 1:

- white pixels, >50% of index peptide response, as 0.75, to classify them as SB,
- yellow pixels, 5%–50% of index peptide response, as 0.25, to classify them as WB,
- red pixels, <5% of index peptide response, as 0.02, to classify them as NB

For a subset of above TCRs, the TCR-pMHC K_d values for mutant peptides are available from the source paper.¹¹⁶ For a subset of these peptides, the K_d values are explicitly denoted in Table 1, which we record directly. For the rest, we extracted the $\Delta\Delta G$ data from the bar heights in Figure 3 and converted them to K_d values using the formula $K_d = \exp(\Delta G / RT)$, with $\Delta G = \Delta G_{WT} + \Delta\Delta G$ with ΔG_{WT} recorded from Table 1, and T assumed to be 298K, with $RT = 0.592$ kcal/mol.

We sourced the TCR sequences from Table S1 of Huseby et al.¹¹⁶ and Supplemental Experimental Procedures of Huseby et al.⁵⁷

MBP-TCR

We recorded the proliferation data of Grogan et al.¹¹⁷ Figure 1A. The index peptides show slightly different values for different replicates, which we averaged over. TCR sequence was not available.

TScan-II TCRs

We recorded the mutational scan data from Figures 2D–2F and 4E peptide fold enrichment data of Dezfulian et al.,⁴⁷ found in Tables S1 and S2 online with the article. For TCRs F5, F13, and F24 we skipped positions 1, 2, 23, and 24 since the epitope sequence is denoted as FRDYVDRFYKTLRAEQASQE.⁴⁷ In the data, peptides sequence replicates are reported as multiple entries, we averaged over all occurrences of each peptide as part of the full 56-aa tile SILDIRQGPKEPFRDYVDRFYKTLRAEQASQEVKNWM-TETLLVQNANPDCKTILKA. For TCR-3598-2, data, peptides sequence replicates are reported as multiple entries, we average over all occurrences of each peptide as part of the full 56-aa tile AMPFATPMEALARRSLAQDAPPLPVPGVLLKEFTVSGNILTIRL-TAADHRQLQLS. We minmax normalized the enrichment data to lie between [0,1] for each TCR separately. TCR sequences for TCRs F5, F13, and F24 are taken from Table S11 of Benati et al.,¹¹⁸ and that of TCR-3598_2 from Table 1 of Poncette et al.¹¹⁹

Additional publications for future consideration

A number of publications report more mutational scan datasets for TCR activation, but the raw data from these papers were not readily available. We plan to include them in our database when they become available. These include for instance Bentzen et al.^{19,120} and Border et al.^{19,120} etc.

TCR activation dataset processing

To normalize single-aa mutant datasets across publications, we scaled TCR activation values by the maximum activation value over all recorded peptides tested against that TCR. The only exceptions to this normalization scheme were for experiments where the TCR activation measurements were on a logarithmic scale (e.g., EC_{50} values), in which case we used the logarithm of the TCR activation values and linearly transformed them to map to the [0,1] interval, as detailed above. Following previous works,¹⁷ we discretized the normalized TCR activation values to 3 ordered levels for downstream classification tasks: no activation ($a_{no} \in [0,0.1)$), weak activation ($a_{weak} \in [0.1,0.5)$), and strong activation ($a_{strong} \in [0.5,1]$). For regression tasks, we directly used the normalized TCR activation

values. More technical TCR-specific notes on data collection and processing, as well as links to source publications, can be found in the Supplementary Notes. A number of publications (see Supplementary Materials for citations) contained further mutational scan experiments relevant for our database, but the associated raw datasets were not readily available to us.

To select the benchmark subset of TCRs (e.g., for Figures 2A and 2D) among all TCRs specific for a common index peptide, we selected the one with the highest balance of TCR activation classes. For this purpose, for each TCR, we calculated the entropy of the frequency distribution of the TCR activation classes, as follows

$$S_{\text{TCR}} = -f_S \log f_S - f_W \log f_W - f_N \log f_N, \quad (\text{Equation 1})$$

with f_S, f_W, f_N being the fraction of strong-, weak-, and non-activating single-aa mutant peptides respectively. For each unique index peptide, we selected the TCR with the highest S_{TCR} . The only exception to this rule was for the index peptide FEAQKAKANKAVD; while all other index peptides had TCR sequences available for either all or none of their specific TCRs, only a subset of FEAQKAKANKAVD-specific TCRs had available sequence. So, for this peptide, we selected the TCR B3K508, with the second highest S_{TCR} among FEAQKAKANKAVD-specific TCRs, since the corresponding TCR sequence, necessary for many TCR-pMHC model predictions in Figure 2A, was available.

Web application for visualizing TCR-pMHC interactions from our database

TCR-pMHC interactions from our database (Figure 1B) are visualized via the web application at <https://batman.cshl.edu/>. All interactive plots are deployed as a RShiny application, using *ShinyDashboard* (v. 0.7.2). The scatter plot displaying peptide clustering based on index-to-mutant distance was generated via *ggplot2* (v. 2.3.4.4) and rendered using *plotly* (v. 4.10.3). The heatmap presenting normalized peptide activation per index peptide was generated with *InteractiveComplexHeatmap* (v. 1.8.0). The Alluvium plot visualizing the binding of index and mutated peptides to TCRs was generated with *ggplot2* and *ggalluvial* (v. 0.12.5). The code for the application is available at https://github.com/meyer-lab-cshl/batman_shiny.

Amino acid hydrophobicity calculation

To calculate aa hydrophobicity, we used the Wimley-White interfacial hydrophobicity scale,³⁴ which is known to best describe TCR-pMHC aa binding characteristics.⁵⁸ This is likely due to the fact that the binding sites of unligated TCRs and pMHC are solvated, and hydrophobic residues at protein-protein interfaces often constitute binding hotspots due to the hydrophobic effect.^{36,121} For aa residues histidine, glutamic acid, and aspartic acid, the above scale provides hydrophobicity of both neutral and charged residues, where we have used the latter. The resulting aa hydrophobicity ordering can be found, e.g., in the horizontal axis of Figure 1F.

MHC binding predictions

For all MHC-peptide pairs in BATCAVE, we predicted MHC binding using the command-line interface of netMHCpan4.1²¹ and netMHCpanII4.3.⁶⁸ We restricted binding predictions to the full length of the peptide sequence and classified binding based on 'Rank_BA' into 'no' [$> 2\%$ rank], 'weak' [$0.5\% - 2\%$ rank], or 'strong' [$< 0.5\%$ rank] binders.

Implementation and usage of pre-trained pMHC-TCR methods

We tested a selection of 15 pre-trained TCR-pMHC methods on our database (Figure 2A). The selection was based on availability of webserver, pretrained models, and ease of installing and running models locally. For all the methods, we used available pre-trained models on our dataset (see also a separate later section for the method *pTEAM*, which we trained and tested on BATCAVE). Each tested method yielded a continuous-valued TCR-pMHC interaction score for each mutant-TCR pair, which was used to calculate 3 AUCs for classification tasks that were subsequently averaged in the final results. The following contains links and summaries of different methods tested, and more technical details on their applications on our database. The project GitHub has input and output TCR datafiles for all 15 TCR-pMHC methods tested in Figure 2A, in https://github.com/meyer-lab-cshl/BATMAN-paper/tree/main/results_batman/paper_figures/figure_2/2a/tcr_pmhc_methods_predictions.

IEDB

For MHCI peptides, we uploaded peptide sequences separately for each MHC to the Webserver <https://nextgen-tools.iedb.org/pipeline?tool=tc1>⁴¹ with prediction model "Class I pMHC Immunogenicity" with default (1,2, C terminal) positions to mask, the corresponding MHC allele selected, and Peptide Length(s) set "as is". Prediction for HLA-B81:01 returned errors, so we could not use them. For MHC II data (binding predictions only), we uploaded peptide sequences separately for each MHC to Webserver <http://tools.iedb.org/mhcii/>⁷⁸ with "Prediction method:" set to "NetMHCIIpan 4.1 EL", the corresponding MHC allele selected, and length set to "as is".

PRIME 2.0

We submitted pMHC information to the webserver at <http://ec2-18-188-210-66.us-east-2.compute.amazonaws.com:3000/>,^{32,70} separately for each MHC allele, and indicated the corresponding MHC allele in the submission form for each submission. We used the output peptide rank ('%Rank_bestAllele' column in the output file) to calculate the AUCs in Figure 2B. The best MHC allele was same as the allele in our database in all cases. No mouse or MHCII predictions were available, only HLAI predictions were available. Among different outputs, we used the PRIME score for predicting TCR activation.

pMTnet_Omni

Prediction is available only for TCRs with at least the CDR3 β sequence. For TCRs with full sequence available, we formatted the data according to https://pmtnet-omni-document.readthedocs.io/en/latest/input_format/index.html (more details in the code to generate input) and used the online webserver of pMTnet_Omni at <https://dbai.biohpc.swmed.edu/pmtnet/analysis-omni.php>⁷¹ for predictions. We used the logit column as prediction. For the TCR2-T TCR with only the CDR3 β sequence available, we used pMTnet (v1), and used the webserver <https://dbai.biohpc.swmed.edu/pmtnet/analysis-base.php>.⁸³

NetTepi

We submitted TCR-pMHC data to the webserver at <https://services.healthtech.dtu.dk/services/NetTepi-1.0/>.⁸⁰ Only HLA-I predictions were available. Among them, no predictions existed for HLA-B81:01. We uploaded peptide sequences separately for each HLA and for 9mers and 10mers to the webserver at <https://services.healthtech.dtu.dk/services/NetTepi-1.0/>, set default values for Relative weight on stability prediction: 0.16, and Relative weight on T cell propensity prediction: 0.10 and indicated the respective peptide lengths and MHC alleles. T cell Propensity score gave better AUC than combined score, so we used the former score.

NetTCR-2.2

The predictions require TCR CDR1 α , CDR2 α , CDR1 β , and CDR2 β sequences, which we sourced from IMGT database for different TRA and TRB genes in data (e.g., from <https://www.imgt.org/IMGTrepertoire/index.php?section=LocusGenesrepertoire=genetablespecies=humangroup=TRAV>). Only MHC-I predictions are available, as mentioned in the webserver. Even though there is no MHC input, the maximum peptide length allowed is 12 (as per the webserver instructions), which is smaller than all MHC-II peptide lengths with associated TCR sequence in our benchmarking dataset. The prediction also needs CDR3 $\alpha\beta$ sequences. We submitted the TCR and peptide sequences to the webserver at <https://services.healthtech.dtu.dk/services/NetTCR-2.2/>⁸¹ with the default parameters Similarity scaling factor: 10 and Percentile rank threshold: 100.

ImRex

We downloaded the source code and pre-trained models from <https://github.com/pmoris/ImRex/tree/master>⁹ and ran them locally, using the same pretrained model and Python command line as specified in the 'Predictions using the pre-built model' section of the GitHub README file. Prediction was available only for TCRs with CDR3 β sequences. We used the output TCR-pMHC prediction score to calculate the AUCs in Figure 2B. While, running the above, we received the following output messages in terminal: "Filtered CDR3 sequences to length: (10, 20)" and "Filtered epitope sequences to length: (8, 11)", indicating prediction restrictions.

ERGO II

We downloaded the source code from the main GitHub link <https://github.com/IdoSpringer/ERGO-II/tree/master>^{74,84} and ran it locally using the same Python command line as specified in the 'Prediction Instructions' section of the GitHub README file. Note that the files Model.py and Predict.py required some changes before they could be run, including renaming a directory "TCR_Autoencoder" in the code to "Models/AE" since the TCR_autoencoder directory was missing. We also added the specification *cpu-only* in load-torch settings. On our benchmarking dataset, the ergo model trained on the McPAS database performed better than the model trained on the VDJdb database, so we reported the former model scores. Prediction was available only for TCRs with CDR3 β sequences.

epiTCR

We downloaded the source code from the GitHub link: <https://github.com/ddiem-ri-4D/epiTCR>⁷⁵ and ran it locally using the same pretrained model and Python command line as specified in the 'Run prediction using the pre-trained model' section of the GitHub README file. Prediction was available only for TCRs with CDR3 β sequences. The code requires MHC allele sequences, which were copied from the sample test file entries ("MHC" column) corresponding to the same alleles, downloaded from <https://github.com/ddiem-ri-4D/epiTCR/blob/main/data/hlaCovertPseudoSeq/HLAWithPseudoSeq.csv.zip>. We used the output TCR-pMHC predicted binding probability ('predict_proba' column in the output file).

As per the authors⁷⁵: "CDR3 β sequences of 8–19 amino acids were provided as TCR input, and peptide sequences of 8–11 amino acids were given as the corresponding peptide." Epitopes restricted to MHC-II and for TIL1383I exceeded this lengths and thus TCR predictions could not be done.

iTCep

We downloaded the source code from the GitHub link: <https://github.com/kbvstmd/iTCep/>¹²² and ran it locally. For peptides without a corresponding CDR3 β sequence, the model predicts a list of possible binding TCRs sequences. We do not use these predictions. The source paper⁷⁷ mentions "the peptide-CDR3 pair sequences were padded to the maximum length of 32" Longer pairs in the input data did not return any result in the output file, so no MHC-II predictions were available.

Attn-TAP

We downloaded the source code from the GitHub link: <https://github.com/Bioinformatics7181/AttnTAP>⁷⁸ and ran it locally. Prediction was available only for TCRs with CDR3 β sequences. The McPAS model performed better than the VDJdb model on our benchmarking dataset, so we reported the former model scores.

ATM-TCR

We downloaded the source code from the GitHub link: <https://github.com/Lee-CBG/ATM-TCR>⁷⁹ and ran it locally. Prediction was available only for TCRs with CDR3 β sequences.

DeepTR

Only HLA predictions were available, of them, no HLA-B81:01 prediction was available. We uploaded peptide data for individual TCRs separately to the webserver at <https://bioinfo.uth.edu/DeepTR/Prediction.php>¹²³ and downloaded the results. Prediction was available only for TCRs with full sequences available.

TCRPrediction

We downloaded the source code from the GitHub link: <https://github.com/kyoheikoyama/TCRPrediction>⁸⁰ and ran it locally. Prediction was available only for TCRs with CDR3 β sequences.

HeteroTCR

We downloaded the source code from the GitHub link: <https://github.com/yuzilan/HeteroTCR>⁸¹ and ran it locally. Prediction was available only for TCRs with CDR3 β sequences and for peptides with length <15.

TITAN

We downloaded the source code from the GitHub link: <https://github.com/PaccMann/TITAN>⁴² and ran it locally. Prediction was available only for TCRs with full sequences available.

Training and validation of BATMAN

Bayesian hierarchical classifier for TCR activation

We first describe how BATMAN works for a given TCR in within-TCR mode. For classification tasks, BATMAN (Figure 2B) performs Bayesian logistic regression to predict the ordered categorical activation level for the given TCR and peptide, $a(\text{peptide}) \in \{a_{\text{no}}, a_{\text{weak}}, a_{\text{strong}}\}$, using the peptide-to-index distance $d(\text{peptide}, \text{index})$ corresponding to the index peptide of the TCR, using this link function¹²⁴:

$$\text{Prob}[a(\text{peptide})|d(\text{peptide}, \text{index})] = \begin{cases} 1 - \text{logit}^{-1}(d_0 - d(\text{peptide}, \text{index}) - c_1), & \text{if } a(\text{peptide}) = a_{\text{non}} \\ \text{logit}^{-1}(d_0 - d(\text{peptide}, \text{index}) - c_1) - \text{logit}^{-1}(d_0 - d(\text{peptide}, \text{index}) - c_2), & \text{if } a(\text{peptide}) = a_{\text{weak}} \\ \text{logit}^{-1}(d_0 - d(\text{peptide}, \text{index}) - c_2), & \text{if } a(\text{peptide}) = a_{\text{strong}} \end{cases} \quad (\text{Equation 2})$$

where the inverse logit function is defined as $\text{logit}^{-1}(x) = \frac{1}{1+e^{-x}}$, d_0 is a constant intercept and c_1 and c_2 are two constant cutpoints with the constraint $c_1 < c_2$, with the following hyperprior distributions:

$$d_0 \sim \text{Normal}(\mu_0, \sigma_0), \quad (\text{Equation 3})$$

$$c_1, c_2 \stackrel{iid}{\sim} \text{Normal}(0, 2), \quad (\text{Equation 4})$$

$$\mu_0 \sim \text{Normal}(0, 2), \quad (\text{Equation 5})$$

$$\sigma_0 \sim \text{HalfNormal}(2). \quad (\text{Equation 6})$$

For any peptide-index sequence pair, the peptide-to-index distance $d(\text{peptide}, \text{index})$ is computed based on position-dependent weights $w(\text{position})$ and a 20x20 aa substitution distance matrix M :

$$d(\text{peptide}, \text{index}) = \sum_{\text{position} \in \{1, 2, \dots, L\}} w(\text{position}) M[\text{aa}(\text{index}, \text{position}), \text{aa}(\text{peptide}, \text{position})], \quad (\text{Equation 7})$$

where each element in $M[\text{aa}(\text{index}, \text{position}), \text{aa}(\text{peptide}, \text{position})]$ corresponds to the substitution of amino acid residue $\text{aa}(\text{index}, \text{position})$ to $\text{aa}(\text{peptide}, \text{position})$ at a given position in the index and peptide sequences, respectively. The diagonal elements of M are all zero, such that the distance from the index peptide to itself is zero. BATMAN infers the weights $w(\text{position})$ and aa distance matrix elements of $M[\text{aa}_1, \text{aa}_2]$.

Position-dependent weights $w(\text{position}) \in [0, 1]$ with $\text{position} \in \{1, 2, \dots, L\}$, where L is the length of the TCR's index peptide, have the prior:

$$w(\text{position}) \stackrel{iid}{\sim} \text{Beta}(\alpha, \beta). \quad (\text{Equation 8})$$

Elements of M follow:

$$M[\text{aa}_1, \text{aa}_2] = D[\text{aa}_1, \text{aa}_2](1 + \delta[\text{aa}_1, \text{aa}_2]), \quad (\text{Equation 9})$$

$$\delta[aa_1, aa_2] \stackrel{iid}{\sim} \text{Normal}(\mu, \sigma) \quad (\text{Equation 10})$$

where D is a pre-defined aa distance matrix (e.g., BLOSUM100) used for constructing the prior for the inferred aa matrix M .

The hyperparameters of $d(\text{peptide}, \text{index})$ have the following, weakly informative hyperprior distributions:

$$\alpha, \beta \stackrel{iid}{\sim} \text{Gamma}(2, 2), \quad (\text{Equation 11})$$

$$\mu \sim \text{Normal}(0, 0.5), \quad (\text{Equation 12})$$

$$\sigma \sim \text{Exponential}(1). \quad (\text{Equation 13})$$

BATMAN is also designed to optionally use pMHC binding information. In such cases, Equation 7 is modified as following:

$$d(\text{peptide}, \text{index}) = \sum_{\text{position} \in \{1, 2, \dots, L\}} w(\text{position}) M[aa(\text{index}, \text{position}), aa(\text{peptide}, \text{position})] + d_{\text{MHC}}, \quad (\text{Equation 14})$$

with the scalar d_{MHC} assuming one of 3 possible values based on NetMHCpan-predicted pMHC binding level:

$$d_{\text{MHC}} = \begin{cases} d_{\text{SB}}, & \text{if pMHC pair is strong binder} \\ d_{\text{WB}}, & \text{if pMHC pair is weak binder} \\ d_{\text{NB}}, & \text{if pMHC pair is non binder,} \end{cases} \quad (\text{Equation 15})$$

with the the prior distribution

$$d_{\text{SB}}, d_{\text{WB}}, d_{\text{NB}} \stackrel{iid}{\sim} \text{Beta}(\alpha_M, \beta_M) \quad (\text{Equation 16})$$

with the hyperparameter prior being

$$\alpha_M, \beta_M \stackrel{iid}{\sim} \text{Gamma}(2, 2). \quad (\text{Equation 17})$$

We verified via prior predictive sampling that these assumptions can yield all anticipated outcomes i.e. activation levels.

Regression tasks with BATMAN

To use BATMAN for regression tasks of predicting continuous-valued normalized TCR activation $a(\text{peptide}) \in [0, 1]$, we modified Equation 2 to

$$\text{Prob}[a(\text{peptide}) | d(\text{peptide}, \text{index})] = \text{Normal}(d_0 - d(\text{peptide}, \text{index}), \sigma), \sigma \sim \text{Exponential}(1), \quad (\text{Equation 18})$$

with all other steps being identical as described above for classification tasks. The performance of such an application is shown in Figure S5A.

Pooling across TCRs for training BATMAN

The hierarchical Bayesian inference set-up allows BATMAN to integrate datasets from multiple TCRs having the same index peptide length ('pooling across TCRs'). In such cases, the positional weight profiles $w(\text{position}, \text{TCR})$ and the intercepts $d_0(\text{TCR})$ are TCR-specific, but have the same prior distributions as specified above, i.e.,

$$w(\text{position}, \text{TCR}) \stackrel{iid}{\sim} \text{Beta}(\alpha, \beta), \quad (\text{Equation 19})$$

and

$$d_0(\text{TCR}) \stackrel{iid}{\sim} \text{Normal}(\mu_0, \sigma_0), \quad (\text{Equation 20})$$

with the hyperparameters α, β, μ_0 and σ_0 having hyperpriors as above. These TCR-specific weight profiles are used to calculate TCR-specific peptide-to-index distances $d(\text{peptide}, \text{index}, \text{TCR})$ similarly as above:

$$d(\text{peptide}, \text{index}, \text{TCR}) = \sum_{\text{position} \in \{1, 2, \dots, L\}} w(\text{position}, \text{TCR}) M[aa(\text{index}, \text{position}), aa(\text{peptide}, \text{position})]. \quad (\text{Equation 21})$$

When we used pMHC binding information, we used the following TCR-specific version of Equation 14:

$$d(\text{peptide}, \text{index}, \text{TCR}) = \sum_{\text{position} \in \{1, 2, \dots, L\}} w(\text{position}, \text{TCR}) M[aa(\text{index}, \text{position}), aa(\text{peptide}, \text{position})] + d_{\text{MHC}}(\text{TCR}), \quad (\text{Equation 22})$$

with the TCR-specific quantities $d_{\text{MHC}}(\text{TCR})$ having the same priors as d_{MHC} before.

TCR-specific peptide-to-index distances are consequently used, similar to Equation 2, to construct TCR-specific activation probabilities $a(\text{peptide}, \text{TCR})$:

$$\text{Prob}[a(\text{peptide}, \text{TCR}) | d(\text{peptide}, \text{index}, \text{TCR})] = \begin{cases} 1 - \text{logit}^{-1}(d_0(\text{TCR}) - d(\text{peptide}, \text{index}, \text{TCR}) - c_1), & \text{if } a_1 \\ \text{logit}^{-1}(d_0(\text{TCR}) - d(\text{peptide}, \text{index}, \text{TCR}) - c_1) - \text{logit}^{-1}(d_0(\text{TCR}) - d(\text{peptide}, \text{index}, \text{TCR}) - c_2), & \text{if } a_2 \\ \text{logit}^{-1}(d_0(\text{TCR}) - d(\text{peptide}, \text{index}, \text{TCR}) - c_2). & \text{if } a_3 \end{cases} \quad (\text{Equation 23})$$

where

$$\begin{aligned} a_1 : a(\text{peptide}, \text{TCR}) &= a_{\text{non}} \\ a_2 : a(\text{peptide}, \text{TCR}) &= a_{\text{weak}} \\ a_3 : a(\text{peptide}, \text{TCR}) &= a_{\text{strong}} \end{aligned}$$

In both within-TCR and cross-TCR cases, pooling was performed over different positions in the peptide sequence, and different elements of the matrix M , corresponding to different aa substitutions. Pooling across aa substitutions allowed us to assign $M[aa_1, aa_2] = D[aa_1, aa_2](1 + \mu)$ for aa substitutions absent in the training set but present in the test set.

Unpooled BATMAN training

We tested different parameter inference and pooling schemes for BATMAN. For all MHCII TCRs, Figure 2E ‘BATMAN unpooled’ results, all Figures 5A and 5C (except ‘pan-TCR’ results) TCRs, Figures S5 and S6, and ‘Full_unpooled’ results in Figure S7, we did not pool across TCRs, i.e., BATMAN was trained individually for each TCR separately. In these cases, the unpooled inferred weights had a Beta(2, 2) distribution as the prior. We first used conventional aa substitution distance matrices, and performed 5-fold cross validation separately for each TCR, e.g., for 9-mer-binding TCRs using about 144 random peptides from the set of single-aa mutants for the TCR for training, and the remaining 36 for testing.

Pooling schemes for BATMAN

In Figure 2D ‘BATMAN (BLOSUM100)’ results, Figures S5 and S6, we did not infer the aa matrix, i.e., M was set to the indicated aa distance matrix. In other cases where we inferred the matrix M , the pre-defined matrix D was always chosen to be BLOSUM100, since it performed the best in classification tasks among all the conventional aa distance functions in unpooled training (Figures S5B and S6). For Figure S7 *Symmetric_** results, we constrained M to be symmetric, which showed slightly worse performance than when we did not impose this constraint. Thus, in all other cases, we inferred the full aa matrix. Indeed, the asymmetric part of the inferred full aa matrix was prominent for hydrophobic aa residues (Figure 3E). For plotting the inferred aa matrix in Figure 3E, we divided the inferred matrix by the corresponding values of $1 + \mu$, to make the ratios of inferred matrix elements to BLOSUM100 matrix elements more interpretable.

When pooling across TCRs, for Figures 2D and 2E ‘BATMAN pooled’ results, we pooled across the selected 14 9-mer-binding TCRs and 4 10-mer-binding human TCRs separately. In Figures 2F, 3B, and S7 **_within* results, we pooled within TCRs specific for an index peptide. In Figures 2D and 2E ‘BATMAN pooled’ results, BATMAN is pooled across TCRs specific for all index peptides of same length in the benchmarking TCR set of Figure 2A, and across all TCRs specific for all index peptides of same length in Figure S7 **_across*. BATMAN performance improved by pooling the training data across TCRs (Figures 2E and S7), even when inferring TCR-specific weights.

We used pMHC binding information in BATMAN prediction for the results in Figures 2D ‘BATMAN’, 2E, 2F, and 3. We did not use pMHC binding information in BATMAN-AL, since LOO-TCR performance using pMHC binding had a broader range across TCRs with using pMHC binding (Figure 2D). For multi-aa mutant results in Figures 5A and 5B, since the experimentally tested multi-aa mutants are biased towards MHC-binders, we did not use pMHC binding information input.

Finally, while in most cases we inferred TCR-specific positional weight profiles, for leave-one-TCR-out tasks (e.g., Figures 2D, 2F, 4A, 4B, and S9 etc) and ‘pan-TCR’ results of Figures 3A and 5C, we inferred a common weight profile for all TCRs in the training set.

Training schemes for BATMAN

For within-TCR validation tasks, we performed 5-fold cross-validation of BATMAN. The folds were stratified by TCR activation levels for classification tasks and TCR activation deciles for regression tasks, and kept identical among all methods (averaged over folds) for comparison.

For TCRs with a sufficient number of peptide examples (≥ 5) of all 3 activation levels to perform 5-fold cross validation, BATMAN classification performance was quantified in terms of 3 pairwise AUCs based on the peptide-to-index distance $d(\text{peptide}, \text{index})$ of each mutant peptide, calculated using TCR-specific or cross-TCR positional weight profile and aa distance matrix inferred by BATMAN. In such cases (e.g., Figures 2D–2F, 4A, 4B, and S5 etc) an average of the 3 AUCs are plotted, whereas Figures S6 and S7 show individual AUCs. For some TCRs, and test folds, only 2 activation classes were available, where the single AUCs corresponding to the only available activation class pair are plotted. For LOO-TCR tests in Figure 2F restricted among SIINFEKL-specific TCRs, we only used educated TCRs (OT1 and Ed TCRs in Figure S2) and discarded low-avidity naive TCRs, for which *pTEAM* performed worse.¹⁷

AA distance matrices in prior distribution

To convert conventional aa substitution matrices (D' set to BLOSUM_*, PAM_*, Dayhoff, or Gonnet) into distance matrices D suitable to be used in priors for BATMAN, we performed the transformation

$$D[aa_1, aa_2] = \left(1 - \frac{D'[aa_1, aa_2]}{D'[aa_1, aa_1]}\right) \left(1 - \frac{D'[aa_2, aa_1]}{D'[aa_2, aa_2]}\right), \quad (\text{Equation 24})$$

so that the AA distance matrix D was always symmetric, with diagonal elements equal to zero. The Hamming matrix had all off-diagonal elements equal to 1.

Parameter inference from posterior sampling for BATMAN

BATMAN is implemented in python (v. 3.11.5), using *pymc* (v. 5.10.1) packages. We sampled inferred parameters from approximated posteriors using the "Automatic Differentiation Variational Inference" (ADVI) method.¹²² More details about the computational implementation of ADVI can be found at <https://www.pymc.io/projects/docs/en/stable/api/generated/pymc.ADVI.html>. The convergence criterion was that the evidence lower bound (ELBO) loss function did not change by more than 0.1% if the number of iterations was doubled. In Figure S10 We show an example of the ELBO loss trace with ADVI steps, as well as traces of inferred TCR-specific positional weight means for a representative BATMAN training which infers the full aa matrix, MHC effects, and TCR-specific positional weights by pooling over 14 9-mer binding TCRs with unique index peptides in our benchmarking dataset (with the inferred weight profiles plotted in Figure 3A). The plots show both inferred parameters ($14 \times 9 = 126$ positional weights) and ELBO loss converge much earlier than the total duration for which we ran the approximation (80k steps). For unpooled BATMAN training on limited subsamples in Figure 2E, we found that we required a minimum of 30% of within-TCR training fold data for the convergence of estimated BATMAN parameters for all TCRs considered.

Active Learning with BATMAN

To initialize parameters, we first trained BATMAN-AL in the leave-one-TCR-out mode, and learned pan-TCR positional weights ($w_{\text{pan-TCR}}$) and the aa substitution distance matrix (M). When applying active learning to a novel TCR, we only modify the TCR-specific positional weights. For all candidate peptides, we used Equation 7 to calculate peptide-to-index distances.

Our strategy to select peptides for training was based on a combination of two criteria: (a) picking peptides near decision boundaries so that classifiers can finely delineate classes, and (b) picking diverse peptides so as to cover different regions of peptide space. For (a), if all peptide candidates (single-aa mutants) were sorted based on their peptide-to-index distance, then assuming balanced data, we reasoned that class boundaries would fall at roughly the median distance (for 2-class data) or the two tertile distances (1/3, 2/3 for 3-class data). For (b), we explored diversity in the positions of the selected peptides, as well as diversity in the distances themselves.

Using these intuitions, our BATMAN-AL strategy was as follows. For each position, we first selected one mutant peptide. For roughly half (5 out of 9) the positions — those with the smallest pan-TCR positions weights — we selected the peptide having the peptide-to-index distance closest to the median peptide-to-index distance. For the remaining (4 out of 9) positions, we selected the peptide closest to the first tertile of the peptide-to-index distances (Figure S9A). To avoid sampling redundant peptides, when selecting more than one peptide per mutation position, we selected the additional peptides randomly.

To infer positional weights for an unseen TCR, we trained BATMAN in within-TCR mode with the sampled peptides and the index peptide as before, except now the priors for the positional weights (w_1 , the positional weight vector for the 1st AL round) are informed by the pan-TCR weights as follows:

$$w_1(\text{position}) \sim \text{Normal}(w_{\text{pan-TCR}}(\text{position}), \sigma(\text{position})), \quad (\text{Equation 25})$$

with

$$\sigma(\text{position}) \stackrel{iid}{\sim} \text{Exponential}(1). \quad (\text{Equation 26})$$

For the next AL round, we sample from the remaining peptides, calculating peptide-to-index distances using the pan-TCR aa matrix and the learned positional weights from the last AL round as a prior:

$$w_i(\text{position}) \sim \text{Normal}(w_{i-1}(\text{position}), \sigma(\text{position})). \quad (\text{Equation 27})$$

After each AL round, we normalized the learned weights to $[0, 1]$, and calculated AUC for the remaining unsampled peptides.

As mentioned above, when selecting more than one peptide per position, our strategy selects the additional peptides randomly. We compared the performance of 4 strategies for selecting random peptides:

1. Peptides are not selected randomly; i.e., peptides are chosen according to their distance from the median or the first tertile of the peptide-to-index distances, depending on the positional weights (i.e., the AL strategy introduced above).
2. All peptides are chosen randomly in each round, while still sampling one peptide per position ("random learning").
3. All peptides are chosen randomly in even rounds, and in odd rounds, peptides are chosen as per strategy (1) above.
4. All peptides are chosen randomly after the first round, with peptides chosen according to strategy (1) in the first round.

For benchmarking performance of AL strategies, among our benchmarking TCR set of Figure 2A, we selected 9-mer-binding TCRs with all positions mutagenized. In all cases involving random peptide selection, we performed AL for 100 realizations of random peptide choice and parameter inference, and averaged the results. The results (Figure S9B) showed that out of the 4 strategies, the 3rd strategy exhibited a monotonic increase in median AUC with the number of peptides sampled, and the smallest variance of AUCs, across different TCRs (Figure S9B). Thus, we plotted the results corresponding to this strategy in the main text (Figures 4A and 4B).

These results also showed the importance of random peptide choice in AL algorithms. For example, AL without any random peptide sampling did not exhibit monotonic performance increase with more experimental rounds (Figure S9B).

Implementation and training of *pTEAM*

A recent method, *pTEAM*, was specifically developed to predict TCR activation by mutants. We implemented *pTEAM* following the description in its source preprint¹⁷ and trained it in both within-TCR and leave-one-TCR-out modes. Briefly, we used Atchley embeddings for index and mutant peptides, and, for leave-one-TCR-out tasks, aligned TCR sequences. These embeddings were used as inputs to random forests with 250 trees for classification and regression tasks, with same folds as BATMAN. Each pairwise AUC was calculated by averaging over two AUCs corresponding to 3 activation level probabilities output from the random forests. We used *R* to align TCR sequences with the *muscle* (v 3.40.0) package and implement the random forests with the *randomForest* (v. 4.7-1.1) package. AUCs were calculated using the *multiclass.roc* function from the *pROC* (v. 1.18.4) package in *R*. All R codes for our re-implementation of *pTEAM* can be found at https://github.com/meyer-lab-cshl/BATMAN-paper/blob/main/results_batman/batman_functions/pTEAM_functions.R.

Assessing the relationship between inferred peptide weights with TCR-pMHC structure

For 6 TCRs in the BATCAVE dataset with full peptide mutational scan, there were corresponding PDB entries with the TCRs in complex with either (1) their respective index peptides in the BATCAVE database (TCRs 47BE7, 1E6, and TIL1383I), (2) a single-aa mutant of their index peptide (TCRs NYE-S with BATCAVE index peptide being SLLMWITQC and structure present for SLLMWITQV), (3) the TCR-binding epitope which is embedded in the longer BATCAVE index peptide (TCRs F5 and F24 with BATCAVE index peptide being FRDYVDRFYKTLRAEQASQE and structure present for RFYKTLRAEQASQ). Since these were the only experimental TCR-pMHC PDB structures available for BATCAVE TCRs, notwithstanding the differences between BATCAVE index peptides and corresponding PDB peptides, we correlated BATMAN positional weights of these TCRs with corresponding structural features with TCR-pMHC interactions.

To compare weight profiles across TCRs, we normalized the positional weights to their TCR-specific maximum. For TCRs F5 and F24, while BATMAN inferred positional weights for all 20 positions of FRDYVDRFYKTLRAEQASQE mutagenized in BATCAVE, PDB structures contained only peptide residues R7-Q19, and thus we restricted our analyses to those for structural analysis.

To correlate the BATMAN inferred positional weights with TCR-pMHC interaction, we analyzed the polar and nonpolar interactions of each peptide residue in the complex with TCR for TCRs with available structural data in the Protein Data Bank (PDB), listed here by TCR name: PDB identifier: NYE-S1:6rpb, NYE-S2: 6rpa, NYE-S3: 6rp9, 47BE7: 7na5, TCR-1E6: 3utt, TIL1383I: 7rk7, TCR-F5: 6cqn, and F24: 6cql. All analyses were performed with UCSF ChimeraX⁸² (version 1.8). To identify hydrogen bonding interactions between residues of the peptide to neighboring residues in either TCR, MHC or peptide itself, we used the 'Hbonds' function which uses atom types and geometric criteria to identify hydrogen bonds. For this, we allowed a distance tolerance of 0.4 Å and an angle tolerance of 20°. To identify all polar and nonpolar interactions involving each of the residues of the peptide to neighboring residues in either TCR, MHC or peptide itself we used the 'contacts' function. For this, we allowed a center-center distance of 4 Å. Threshold parameters for interactions represent default cut-offs for the range of the respective interaction.¹²⁵ For each pMHC-TCR structure, we then summed the number of polar and non-polar interactions at each peptide position and fit a linear model with interactions as the independent and BATMAN-inferred peptide positional weight as dependent variable.

ADDITIONAL INFORMATION

Any additional information required to reanalyze the data reported in this paper is available from the [lead contact](#) upon request.