Autoencoder based local T cell repertoire density can be used to classify samples and T cell receptors

T cell repertoire density estimate

Shirit Dvorkin¹, Reut Levi¹, Yoram Louzoun^{1, +}

1. Department of Mathematics, Bar Ilan University, Ramat Gan, Israel.

+ To whom correspondence and requests should be sent: louzouy@math.biu.ac.il

Abstract	10
Recent advances in T cell repertoire (TCR) sequencing allow for characterization of repertoire	11
properties, as well as the frequency and sharing of specific TCR. However, there is no efficient	12
measure for the local density of a given TCR. TCRs are often described either through their	13
Complementary Determining region 3 (CDR3) sequences, or their V/J usage or their clone size.	14
We here show that the local repertoire density can be estimated using a combined representation	15
of these components through distance conserving autoencoders and Kernel Density Estimates	16
(KDE).	17
We present ELATE – an Encoder based Local Tcr density and show that the resulting density	18
of a sample can be used as a novel measure to study repertoire properties. The cross-density	19
between two samples can be used as a similarity matrix to fully characterize samples from the	20
same host. Finally, the same projection in combination with machine learning algorithms can	21
be used to predict TCR-peptide binding through the local density of known TCRs binding a	22
specific target.	23
Code availability- https://github.com/louzounlab/Autoencoder	24
Keywords: T cell receptor, Local density, Autoencoder, Distance conserving, Machine learning	25
	26

Introduction	27
T-cell receptor (TCR) repertoire formation results from two major procedures: Rearrangements	28
of the TCR α and β chains gene segments (V (D) and J) and formation of the junctions between	29
them [1]. During segments joining, nucleotides are inserted and deleted at the junctions between	30
pairs of rearranging genes, forming the highly variable complementarity determining region 3	31
(CDR3) that is directly implicated in antigen recognition, and further augmenting diversity [1,	32
2].	33
Repertoire next-generation immune-sequencing methods (often denoted rep-seq)[3] supply	34
detailed measures of T cell clone frequencies in samples[4-8]. At the broad level, Rep-Seq	35
methods have three main usages: A) Descriptive - general statistics of the repertoire, including	36
clone size distribution, V, D and J usages, CDR3 length, descriptions of alleles and haplotypes,	37
and in B cells also properties of somatic hypermutations and clonal trees[3, 9]. B) Comparative	38
- comparison between different hosts or different compartments within the same host and the	39
sharing of clones between such compartments, comparison between repertoires in different	40
conditions (e.g. patients with specific diseases), and even comparisons between B and T	41
cells[10, 11]. C) Predictions and Machine learning, including prediction of receptor targets, and	42
prediction of compartments using CDR3 composition or V gene usage [12].	43
The full sequence of T cell receptors is typically not studied. Instead, sequences are	44
characterized by their V and J genes, their observed clone size, and CDR3 sequence[13]. The	45
CDR3 sequence, the V-beta usage, and the clone size distribution are usually analyzed as	46
distinct features of the repertoire[2, 14, 15]. The separation between these components is mainly	47
the result of the different representations (categorical for V(D)J usage, CDR3 sequence, and	48
counts for clone size). There is a need for an efficient framework to unify them into a single	49
coherent representation. Moreover, while there are standard methods to compare V(D)J usage	50
(e.g. Kullback Leibler divergence on the distributions [16]) or the clone size distribution (e.g.	51
entropy or Simpson index [14]), an efficient measure to define TCR local density based on the	52

CDR3 sequence is still needed. The local density of a TCR can be defined through the distance 53
of a TCR to other nearby TCRs. Such a density, requires a definition of a distance between 54
TCRs - for example the edit distance between CDR3 sequences. However, computing the edit 55
distance between each pair of CDR3 sequences is prohibitively expensive. for large samples. 56

Estimating the TCR local density in the receptor space has important applications, including 57
the comparison of densities of different TCR groups, estimating the cross density of different 58
repertoires (i.e. the density of one TCR from one repertoire around TCRs from a different 59
repertoire), and characterization of the TCR distribution properties. For example, in order to 60
follow the developmental stages of T cells, one may want to detect regions in receptor space 61
that are evolving to become denser or less dense. Similarly, one can ask whether some regions 62
are “holes” in the repertoire, with a very low receptor density. We here propose that through 63
autoencoders and kernel density estimators, CDR3 sequence, V/J gene and clone size can be 64
unified to provide a computationally efficient description of the repertoire, and the local density 65
in receptor space around each TCR can be properly defined. 66

Autoencoders are typically used for dimensionality reduction[17]. An autoencoder is an 67
encoder-decoder combination where the output of the decoder is simply the input to the 68
encoder, and the goal is to minimize the difference between the two. Autoencoders are used to 69
design efficient data representation. Encoders project an input vector to a lower dimension. The 70
decoder projects the low-dimensional vector back to the original dimension. In neural nets 71
based autoencoders, during the training process, the autoencoder minimizes the difference 72
between the input of the encoder and the output of the decoder. The low-dimensional projection 73
is then used as an efficient representation of the input vector. 74

We propose the following outline for repertoire characterization by developing ELATE 75
(Encoder based Local Tcr dE nsity), with each step further explained in the Results and 76
Methods sections: 77

- A) Use an autoencoder (either distance maintaining or not) to project each CDR3 sequence into a Z dimensional real space. 78 79
- B) Project repertoires using a constant number of clones from each repertoire. 80
- C) Define cross/self repertoire local density in this Z-dimensional space using KDE methods[18]. 81 82

To test the possibility of estimating the local density of T cell receptor sequences as defined by their CDR3 sequence, their V gene, and their clone size, we developed the ELATE encoder. This analysis is mostly focused on the beta chain since it has been assumed the repertoire diversity and its binding to target MHC-peptide are determined more by the beta chain than the alpha chain[19], and there are more beta than alpha-beta published datasets. However, ELATE can be equally applied to the alpha chain. 83 84 85 86 87 88

Results 89

ELATE can produce accurate reconstruction and also accurate TCR distances. 90 91

ELATE projects the CDR3 sequence and optionally the V gene using a Multi Layer Perceptron (MLP) based AE. (see Methods and Fig. 1 for encoder structure and for the computational flow for a single receptor). An Long-Short-Term-Memory (LSTM) recurrent neural network [20] based approach to handle different lengths was also tested but had a lower decoder accuracy. Similarly, when comparing encoder and LSTM for predicting TCR-peptide binding, encoders provided better results[21]. The decoder predicts the entire input vector, including the stop signal and what comes after it. All values after the predicted stop signal are then zeroed. To evaluate the performance of the AE, we used the percentage of sequences successfully reconstructed with up to K errors, where a mismatch in length was defined as an error for all K s. Note that the current definition does not take into account the similarity between amino acids as measured through either their evolutionary distance or through their biochemical 92 93 94 95 96 97 98 99 100 101 102

distances. Such distances can be easily integrated in both the training and evaluation stages of ELATE.

In all the observed datasets, very few pairs of CDR3 sequences differ by less than 2 AA one from each other (more precisely have an edit distance of less than 2). Indeed, when the distribution of edit distance between every pair of sequences from pairs of random samples is calculated, the distributions are consistently normal, with practically no values below 2 (see a representing example in Fig. 2A). Thus, even an AE producing 1 or 2 errors would lead to a representation closer to the original CDR3 than the vast majority of other sequence in a different sample. Note that the absolute number of non-shared CDR3 sequences within 2AA of a CDR3 sequence from another sample depends on the sampling depth. We thus report the fraction of sequences successfully reconstructed with 0, 1, or 2 errors (Fig. 2).

To test the applicability of the AE, several previously published datasets with different categories were used in this analysis. The first set studied was the GLIPH dataset which is composed of seven TCR samples, where all TCRs in a sample share specificity to the same peptide[22]. Two more datasets of tumor-bearing mice from ImmunoMap and TILs[23] were used. The first had two categories: CD8 T cells responding to TRP2, a shared self-peptide tumor antigen and SIY, a model foreign-antigen. The second is composed of samples taken from mice treated with RT and 9H10 immunotherapies and a combination of both. Also, a set of sorted naïve and memory cells (Benny Chain dataset), CD4+ and CD8+ from alpha and beta chains of three donors was collected[24]. To explore variations over time, we used the response to the yellow fever vaccine from different time-points in three pairs of identical twins[25]. Finally, a collection of TCRs with one sample from each host with any type of cancer and samples from healthy donors were collected from the Immune Epitope Database (IEDB). The cancer analysis was performed to show the applicability of ELATE to TCR samples not directly taken from deep sequencing experiments. The GLIPH and Benny Chain datasets do not contain V usage in their repertoire and the GLIPH dataset has no frequencies of clones.

While ELATE can quite precisely encode the CDR3 distribution (Fig. 2C, D), it fails to reproduce the distance between TCRs (Fig. 2E). To handle that, we produced a novel distance maintaining encoder (EM). In this framework, besides minimizing the reconstruction error. An embedding loss ($loss_{DIS}$) of the sum over pairs of TCRs of the difference between the Euclidean distance between the embeddings and the original Euclidean distance between the CDR3 one-hot vectors is added at the hidden layer. When the V gene of each TCR was provided, a V gene-specific E was also concatenated to the CDR3 encoder. The V based encoder can be used with or without the embedding loss.

The accuracies with zero, one, and two mismatches for all four models (E, VE, EM, VEM) are shown for all the test set of all datasets (except for GLIPH and Benny Chain when analyzing V) in Figures 2C-D, G-H. None of the receptors in the test set were used for either training or hyper-parameter tuning. The accuracy of E is consistently much higher than the EM, with or without V, with high accuracy of >90% in multiple data sets for 2 mismatches accuracy. The models with the additional embedding constraints have lower accuracy (Fig. 2G, H), as expected given the need to minimize in parallel two loss functions. Similarly, encoding with V gives a lower accuracy than without it, again as expected from the larger number of degrees of freedom. However, as will be further shown, the encoding with the embedding is better for multiple applications. To test whether the addition of the distances loss does maintain the distance distribution, the pairwise edit distances of the input one-hot vectors of the CDR3s in the original set and the all distances in the new representations set were compared (Fig. 2E, F). The correlation is higher in the constrained model (Fig. 2F) with a Spearman correlation of 0.9 ($p < 1.e-100$), compared with the regular AE (correlation of 0.5, Fig. 2E).

ELATE can be used to detect features of single clones.

To highlight the importance of the encoding, we first tested whether the encoder reveals the reported selection against specific AA in CDR3 sequences[26]. Samples composed of 100 TCR per sample from the Benny Chain dataset were projected using the AE. To properly visualize the projection, TSNE was used to lower the dimensions to 2 (Fig. 3A). Although we used TCR

from different samples, consistently, TCRs with Cysteines (red color) were projected as clusters (Fig. 3A). Moreover, CDR3 sequences containing a Cysteine or Glycine have lower densities than other CDR3 sequences (Fig. 3G). Cysteines form disulfide bonds, an important feature that contributes to the stability of proteins, yet in CDR3, such bonds may deform the CDR3 surface. Therefore, most CDR3 sequences do not contain Cysteines. For Proline, we observe an interesting difference, where TCRs with a single proline are denser than either without or with 2. We currently have no clear explanation for that.

Next, we tested whether the AE captures the CDR3 length distribution of the repertoire. Indeed, if distances are maintained, similar length CDR sequences should have close projections. Moreover, CDR3 with much longer or shorter than average sequences should be again projected to the periphery, as they are also less frequent and selected against[26, 27]. Again, the AE projects the mid-length CDR3 sequences to the center of the distribution and the long and short CDR3 sequences to the periphery (Fig. 3B, where each color is a different length).

Public TCRs have been described in a variety of immune responses[28]. A public clone is a clone shared by different donors. Again, if the encoded projections represent the functionality of the repertoire, one would expect public clones to be more central. We calculated the radius from the center of mass of all projected TCRs in Benny Chain dataset. Note that such an analysis is only possible following the translation to real-valued representations. We computed the radius of TCRs vs their publicity, where the publicity of each sequence is defined as the number of samples containing it, out of 56 samples (Fig. 3C). Indeed, public clones are more central (spearman correlation, $p < 0.001$). We further tested if public clones are in denser regions of the projection, where the self-density was defined as the density of other TCR projections from the sample around the clone. Density was estimated using a Kernel Density Estimation (KDE). Again public clones reside in denser regions than private clones (spearman correlation, $p < 0.001$) (Fig. 3C, D).

Finally, we checked the overall characterization of the TCR repertoire as a possible biomarker 181
for different diseases[29], for example, a biomarker for cervical cancer[11]. We expect TCR 182
samples (those are not actual rep-seq, but rather samples of reported TCRs) in such conditions 183
to show distinct projections compared with random repertoires. To evaluate the scattering of 184
different categories in the embedded space, we used the AE projections in two disease state 185
datasets. Again, the vectors were projected to 2 dimensions using TSNE to visualize their 186
positions. When analyzing the GLIPH data set, all of the clones in the TPRVTGGGAM peptide 187
are significantly clustered together (Fig. 3E). In the cancer data set, when projecting the Cancer 188
vectors to 2 dimensions, a separation between most TIL and naïve TCRs emerged (Fig. 3F, 189
and t-test with the y-axis position $p < 0.001$). In contrast with the previous tests, the cancer 190
samples are from the IEDB and are enriched for disease specific TCRs. One can now use this 191
density to estimate whether peptide-specific TCRs are in denser regions than random peptides 192
from the same sample. For almost all targets, TCRs specific for this target have a higher density 193
than random TCRs, not defined to bind a known target. This could be a sampling bias or a real 194
difference, but the current method is the first that allows for such a measurement (Fig. 3G). 195

ELATE with KDE can be used to cluster samples. 196

The autoencoder can be used to compare samples and not only to compare TCRs within a 197
sample. Again, we used a constant sample size (100 here) to avoid sample size bias. We then 198
defined a similarity between samples and tested whether this similarity can be used to detect 199
which sample comes from the same host. The similarity was defined using the AE, the 200
Kullback- Leibler (KL) divergence of the V gene usage on the sample, or the edit distance (ED) 201
between CDR3 of different samples. The ED distance between each sequence from one sample 202
and each sequence from the other was computed. The distance between the two samples was 203
defined as the average of all minimum ED values (see Methods). For the AE based distance, 204
the similarity between samples A and B is defined as the average KDE based density of sample 205
A projected sequences based on the projected sequences of sample B. The distance was 206
computed for different variants of the AE, including the basic autoencoder (E), the autoencoder 207

combined with V (VE) or with CDR3 distances (EM) and combined with both (VEM). The KDE was performed twice, one time with the original set of vectors, and again when the clones frequency was incorporated (see Methods). We analyzed the vaccine dataset. This dataset contained samples from three pairs of identical tweens. The samples were taken at six different time points before and after a yellow fever vaccination. Hierarchical clustering was performed using the distance/similarity definitions above. The clustering that best reproduces the division to hosts is VE (Fig. 4A). Moreover, twins are closer to each other in the hierarchy than unrelated donors. When computing the distances using only the V usage (KL distance), the clusters are shuffled, with some distinction of twins but no clear separation of hosts (Fig. 4B). To quantify the distance method that best characterizes hosts, we tested the hierarchical clusters host ID frequency and time-points entropies for each method (Fig. 4C, D). All AEs produce a better detection of hosts (lower entropy) than ED and KL methods, and the basic autoencoder combined with V (VE) without the frequencies is the best method among all models (Fig. 4C). However, none of the methods separates samples by time-points (Fig. 4D).

ELATE with KDE can be used for supervised sequence classification.

Another important application of AE is sequence classification. A classifier was built on top of the AE. The classifier is a combination of the encoder with three fully connected layers, which replace the decoder (Fig. 1). After the autoencoder was trained to predict the vector representation of the CDR3 set, the classifier layers are trained using a binary cross-entropy loss on the class detection problem. Such a classifier was built and trained for each of our four autoencoders. Each classifier was trained as one vs all classifier (i.e. for each class, this class vs all others), for each of the categories (Table 1). All test set AUC values are shown in Table 2.

We compared the first two datasets with DeepTCR results for a single sequence classifier. When analyzing GLIPH and Benny Chain datasets, no Vs are provided, so there are no models with the combination of V. As for the last three datasets (Benny Chain, vaccine and cancer), we have no comparison since there are no previous classifications for these repertoires. Deep-

TCR's based classifiers typically outperform ELATE on GLIPH. However, ELATE based
 classifier outperforms the Deep-TCR's sequence classifier when classifying TRP2 and SIY
 samples (ImmunoMap) with an average AUC of 0.81 vs 0.71. ELATE also performs best when
 handling TCR samples taken from healthy hosts and hosts with cancer, with an average AUC
 of 0.925. Here, different combinations of the autoencoder show different results in different
 datasets and there is no clear pattern or preference for any model.

Self-density estimate using ELATE

Finally, we checked whether the AE can be used to determine how dense is a given sample.
 This can be defined by the average similarity between TCR in a dataset, or alternatively the
 frequency of other TCRs from the same dataset around each TCR. We analyzed again the
 Benny Chain data set of TCR repertoires, and compared alpha to beta chains, naïve to memory
 and CD4 to CD8 cells self-densities. The self-density of a given sample S is formally defined
 as the KDE based density between S and itself, where each TCR is ignored for the computation
 of the density at its own projection. The average density of all groups was normalized to 0 over
 all projections for the ease of presentation (Figure 5).

We computed the log of the self KDE values among all samples for the two autoencoders (E,
 EM), with or without the frequencies, alpha chains have a lower self-density than beta chains.
 This is surprising given the theoretically much larger beta repertoire. When the embedding is
 used, a clear hierarchy can be observed alpha is less dense than beta, CD4 is on average less
 dense than CD8 and naïve are slightly less dense than memory. In the encoder or when the
 clone size is used, the same order is kept, but with a much noisier separation. Thus, again the
 embedding seems to be the method to represent receptors as real-valued vectors.

Discussion

Characterization of the repertoire can serve as an indicator for immune monitoring and
 prognosis assessment for different diseases[11]. Next-generation sequencing (NGS) allows

massive analysis of the TCR repertoire with high-throughput and provides descriptive and
comparative analysis for million TCRs. The repertoire can analyzed through different features
such as the CDR3 set or V gene usage and more. However, current methods of analyzing the
repertoire are focused on one measure or another. We have here proposed ELATE a method to
integrate them. ELATE, is a projection of TCRs that combines different aspects of the repertoire
into a coherent real-valued representation of each clone. ELATE conserves important features
of the repertoire. It can be used with unsupervised and supervised methods to distinguish
different groups and assign a sequence to a group. ELATE is highly efficient and can be applied
to repertoires with millions of clones using standard hardware.

Representing clones as vectors in real space has many advantages, including the possibility of
measuring the repertoire density to detect sparse vs dense regions, computing the similarity
between repertoires, using clones for machine learning applications, which are often based on
real-valued input and visualizations. We have here shown that ELATE can indeed perform
these different tasks, and provide important insight on the repertoire, including among many
others:

- A) The effect of amino acids in the CDR3 region on the repertoire density (with a
decreasing density as a function of the number of Glycines or Cysteins), or the
increased density around TCR recognizing cognate antigens.
- B) The possibility of classifying single clones to specific cognate antigens or specific
compartments.
- C) The estimate of the repertoire density and the difference between the density of the
repertoire in naïve and memory compartments.
- D) The similarity between repertoire and showing that samples from different
compartments and time points of the same donor are much more similar to other
samples from the same donor than to any other sample. Samples of twins are much
more similar than samples of non-twins.

While we have shown some typical applications of ELATE, the range of possible applications 286
is much greater. Each flavor of ELATE has advantages and limitations. Distance maintaining 287
versions are better for density estimates but produce worse sequence reproduction. Adding the 288
V gene to the encoder improves the quality of clone classification but further reduces the 289
reproduction accuracy. Incorporating the clone frequency helps to distinguish between the 290
average density of some compartments, but often ruins the distance definition between 291
compartments. Thus, for each application of ELATE, one must consider which element should 292
be introduced. 293

We have used standard parameters for ELATE to ensure optimal reconstruction in the vanilla 294
flavor of ELATE (AE). One may further improve the accuracy of different tasks by tuning the 295
projection parameters for this specific task. However, we suggest that using a uniform 296
projection within and between experiments is important to allow for a consistent method to 297
compare repertoires among different settings. 298

A previous variational autoencoder (VAE) has been proposed in the literature - DeepTCR[12]. 299
The autoencoder used here is different than the VAE used in DeepTCR. The main differences 300
between ELATE and DeepTCR are that in ELATE all CDR3 sequences are converted to one- 301
hot vectors as opposed to DeepTCR, where the sequences are converted to vectors using a 302
trainable embedding layer. Given the wide CDR3 length distribution, ELATE uses right zero- 303
padding to combine different lengths, and a novel loss function to optimize the projection. 304
Finally, and most crucially, ELATE conserves distances, in contrast with all existing methods, 305
allowing for the usage of KDE methods on the projection. At the technical level, ELATE uses 306
only fully connected layers, while DeepTCR uses convolutional layers. 307

308

Methods	309
Studied Samples	310
Six previously published datasets with different categories were collected to test ELATE (Table 3):	311
• <u>GLIPH (Glanville) [22]</u> . TCR sequences were taken from T-cell binding to 7 different target peptides. Each peptide has one sample of TCRs with a high probability of sharing specificity to it. The TCRs from different individuals were clustered using the GLIPH algorithm.	312
• <u>ImmunoMap (Sidhom) [23]</u> . CD8 T cells responding to Kb-TRP2, a shared self-peptide tumor antigen, and Kb-SIY, a model foreign-antigen, in naïve and tumor-bearing B6 mice.	313
• <u>TILs (Tumor-infiltrating lymphocytes) Immunotherapies (Rudqvist) [30]</u> . Samples were taken from cohorts of tumor-bearing mice treated with radiotherapy (RT), anti-CTLA-4 (9H10) and a combination of them.	314
• <u>Benny Chain[24]</u> . Peripheral blood mononuclear cells (PBMC) of three donors were sorted into memory (central memory, CM; effector memory, EM; effector memory RA-expressing revertants, EMRA) and naïve CD4+ and CD8+ populations, alpha and beta chains.	315
• <u>RF Vaccine[25]</u> . Yellow fever vaccine (YFV) responding TCRs in three pairs of identical twin donors (P1, P2, Q1, Q2, S1, S2) at different time-points.	316
• <u>Cancer[31]</u> . A collection of TCRs from human hosts with any cancer and healthy hosts downloaded from the Immune Epitope Database (IEDB) (See Github for fasta file of healthy and cancer TCRs)..	317
• Not all datasets contain resolved Vs or clone frequencies. All categories, number of samples, and whether V and frequencies are available are summarized in Table 3 All datasets are available at https://github.com/louzounlab/Autoencoder .	318

	335
Data Curation	336
TCR sequencing files were collected from the sources above. The samples were parsed to	337
compositions of CDR3 Amino Acid sequences and when available the V usage and the	338
frequency per clone. Sequences with non-IUPAC letters (X, *, _, #) were ignored. For each	339
dataset, we used the pre-processing performed in the original manuscript. The J usage was	340
ignored here for the sake of clarity. It can be incorporated the same way as V.	341
Glossary	342
Table 4 is a detailed glossary of all technical terms used in the analysis.	343
Data Preprocessing	344
The full flow of analysis is described in figure 1. Since TCRs have varying lengths of CDR3	345
sequences, right zero-padding was added to all TCRs to generate a constant length	346
representation. A stop signal (!) was first added at the right end of each sequence of length l .	347
The sequence was then converted to a one-hot vector. Each character had a corresponding 21	348
dimensions one-hot vector (for 20 possible amino acids (AA) and an additional position for the	349
stop signal). The one hot was right zero-padded to produce a constant length representation	350
(Fig. 1). Note that other alignment options could be considered, such as performing a multiple	351
sequence alignment on the CDR3 sequences before the encoding. However, the proposed	352
methods provided the highest accuracy on average on the tested samples.	353
When combining V usage with the CDR3 sequences, the V genes were represented as n_V	354
dimensional one-hot vectors, where n_V is the number of all possible Vs in a given dataset (each	355
dataset has a different number of different recognized Vs based on the algorithm used for the	356
clone analysis and the sequencing length). The CDR3 and V one-hot vectors were then	357
concatenated. The final representation is the concatenated one-hot vectors in this order: AA ₁ ,	358
AA ₂ , ..., AA _i , stop codon, zero-padding, and (optionally) V. The clone size was never used for	359
the training of the autoencoder.	360

Autoencoder (AE) training 361

The AE of ELATE has three internal fully connected symmetric encoder and decoder layers of 362
300, 100, and 30 nodes. Each layer has a dropout rate of 0.1 and an 'Elu' activation function. 363
Beyond that, there are three additional layers in the decoder (Fig. 1A): 364

- A) A reshape layer (to $n/21 \times 21$) that only changes the output vector to a matrix, where 365
each row represents a position in the CDR3, or the V gene (Technically, the V is 366
represented as different vectors, since it has a different dimension than the AA). 367
- B) A fully connected layer with 'softmax' activation function on each row separately 368
- C) Another reshape layer (back to $n \times 1$ dimensions). 369

The AE was trained with an 'Adam' optimizer with a learning rate of 0.0001 and a Mean Square 370
Error (MSE) loss (Eq. 1). The number of epochs for the training varied for different datasets, 371
where N is the number of samples in the training set. 372

$$1.loss_{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad 373$$

Training AE to maintain the distance 374

To combine the original CDR3s distances in the training process, we first computed the $N \times N$ 375
size distance matrix D_N of all Euclidean distances among all pairs of one-hot representation of 376
TCRs from all samples in the training set. Then, batches of b samples were generated from the 377
training set along with their corresponding b size distances matrices D_b produced from D_N . 378

Then, to conserve the distances in the embedded space, the Euclidean distances between pairs 379
of projections and the original Euclidean distance between the original pairs in the input set 380
(Eq. 2). was added to the loss function ($loss_{DIS}$). 381

$$2.loss_{DIS} = \sum_{i \in b} \left(\sqrt{\sum_{j > i \in b} (z_i - z_j)^2} - D_{ij} \right)^2 \quad 382$$

where z_i and z_j are the embedded vectors of x_i and x_j and D_{ij} is the Euclidean distance between 383
 x_i and x_j . 384

Autoencoder accuracy 385

To test the AE prediction, we used the fraction of sequences properly reconstructed, instead of 386
the loss function used for the learning process (See figure 1B). The reconstructed sequence was 387
produced by replacing the softmax by a rigid max in layer B above, and a translation of the 388
one-hot back to a sequence, stopping at the leftmost stop signal or at the end of the CDR3. All 389
positions beyond the leftmost stop signal were ignored. If the length of the reconstructed vector 390
was different than the length of the original vector, the entire reconstruction was defined as an 391
error. Otherwise, the error level was defined as the number of positions differing between the 392
original and reconstructed sequence. Thus an error rate of 15 % for one mismatch implies that 393
15 % of all sequences either had a wrong length or the proper length and one mismatch. 394

Training Classifier 395

The classifier is composed of the previously described encoder, combined with four fully 396
connected layers with 30, 15, 10 and 1 dimensions, with a dropout of 0.1 after the second and 397
third layers, and a *Tanh* activation-function except for the last layer, where we use a sigmoid. 398
The classifier is trained with similar parameters as the encoder and a cross-entropy loss (Eq. 3) 399
(Fig. 1A). When the classifier was trained, the encoder weights were fixed. For multi-class, the 400
encoder was trained as one vs all. 401

$$3.loss_{BinaryCrossentropy} = -\frac{1}{N} \sum_{i=1}^N y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i)) \quad 402$$

The classifier was trained as one vs all classifier, for each of the categories in the various 403
datasets. 404

The TCRs were divided into a training fraction composed of 500 TCR CDR3 sequences from 405
each sample and all the remaining CDR3 sequences (the vast majority) as a test fraction. 406
Hyperparameter tuning was performed using a single dataset and the E encoder. 407

KL Similarity 408

When analyzing only V gene usage, the V-gene distributions were computed for a random set 409
of R crsprinnlones from each sample. Then, the distance between the distribution of a pair of 410
samples p and q was defined to be their KL divergence [32] (Eq. 4). 411

$$4. D(p \parallel q) = \sum_{v_i \in V} p(v_i) \log\left(\frac{p(v_i)}{q(v_i)}\right) \quad 412$$

ED Similarity 413

A two-sample edit distance was determined as the average of all minimum values per row in E , 414
where E is a $R \times R$ distance matrix of the edit distances (Levenshtein distance) of R random 415
clones in both samples. Levenshtein distance represents the number of insertions, deletions, and 416
substitutions required to change one word to another²⁹. When computing the self-distance, the 417
diagonal values were ignored. Note that the edit distance is different than the one-hot distance. 418

KDE Similarity 419

Given two of projections of samples, $S1 = \{x_1, x_2, \dots, x_R\}$ and $S2 = \{y_1, y_2, \dots, y_R\}$ sprins[sprins 420
(or a single projection if $S1$ and $S2$ are the same), we estimated the local density using a Kernel 421
Density Estimator KDE[18]. However, in contrast with standard KDE, we here found the 422
density of each projection in $S1$ with respect to the projections in $S2$ (Eq. 5). 423

$$5. f_h(x) = \frac{1}{Rh} \sum_{i=1}^R K\left(\frac{x - y_i}{h}\right) \quad 424$$

where K is the kernel, and $h > 0$ is a smoothing parameter called the *bandwidth*. x is the 425
projection of a TCR in sample $S1$, while y are all projections in sample $S2$ 426

We can define the average density of the projections $S1$ in $S2$ as the average of densities for all 427
vectors in $S1$. When measuring the density of samples in $S1$ vs $S2$ in combination with the 428
frequencies, each TCR clone was multiplied by its clone size (Eq. 6). 429

$$6. f_h(x) = \frac{1}{Rh} \sum_{i=1}^R freq_{x_i} K\left(\frac{x - x_i}{h}\right) \quad 430$$

Also, the new representation of the data allowed us to define for the first time the average self- 431
density of a given repertoire, as the density when S1 and S2 are the same, and ignoring for each 432
projected sequence its own contribution to its density. Here, we used Gaussian kernel and $h =$ 433
 $1.06 \cdot \text{STD}(S) \cdot \text{len}(S)^{-1/5}$ [33]. 434

Statistical analysis 435

The difference between the various methods in our analysis was tested using two-way 436
ANOVA[34],and Tukey's test. An ANOVA test was performed with all autoencoder models. 437
Then, Tukey's test was performed between all methods and categories pairs in order to find 438
which are significantly different. The clustering of samples was done using Hierarchical cluster 439
analysis (HCA) [35] using average linkage on the KL, ED and KDE distances matrices. 440

Visualization with TSNE 441

T-distributed Stochastic Neighbor Embedding (TSNE)[36] is a nonlinear dimensionality 442
reduction technique well-suited for embedding high-dimensional data for visualization in a 443
low-dimensional space of two or three dimensions. Specifically, it models each high- 444
dimensional object by a two- or three-dimensional point in such a way that similar objects are 445
modeled by nearby points and dissimilar objects are modeled by distant points with high 446
probability. We here use TSNE to visualize the embedded vectors of 30 dimensions in two 447
dimensions. 448

Acknowledgements 449

This manuscript has been released as a pre-print at bioRxiv, : 450
<https://www.biorxiv.org/content/10.1101/2020.06.12.148502v3>[37] 451

452

453

References

1. Bassing CH, Swat W, Alt FW. The Mechanism and Regulation of Chromosomal V (D) J Recombination. *Cell*. 2002;109(D):45-55. 455
2. Benichou JIC, van Heijst JWJ, Glanville J, Louzoun Y. Converging evolution leads to near maximal junction diversity through parallel mechanisms in B and T cell receptors. *Physical Biology*. 2017;14(4):045003-. doi: 10.1088/1478-3975/aa7366. 456
3. Benichou J, Ben-Hamo R, Louzoun Y, Efroni S. Rep-Seq: uncovering the immunological repertoire through next-generation sequencing. *Immunology*. 2012;135(3):183-91. doi: 10.1111/j.1365-2567.2011.03527.x. 457
4. Heather JM, Ismail M, Oakes T, Chain B. High-throughput sequencing of the T-cell receptor repertoire: pitfalls and opportunities. *Briefings in bioinformatics*. 2018;19(4):554-65. doi: 10.1093/bib/bbw138. 458
5. Muraro PA, Robins H, Malhotra S, Howell M, Phippard D, Desmarais C, et al. T cell repertoire following autologous stem cell transplantation for multiple sclerosis. *The Journal of clinical investigation*. 2014;124(3):1168-72. doi: 10.1172/JCI71691. 459
6. Kedzierska K, Izraelson M, Tatiana O, Egorov ES, Kasatskaya SA, Ekaterina V, et al. Origin and evolution of the T cell repertoire after posttransplantation cyclophosphamide. 2016;9(June):133-44. doi: 10.3389/fimmu.2018.01453. 460
7. Pogorelyy MV, Fedorova AD, McLaren JE, Ladell K, Bagaev DV, Eliseev AV, et al. Exploring the pre-immune landscape of antigen-specific T cells. *Genome Medicine*. 2018;10(1):68-82. 461
8. Gerlinger M, Quezada SA, Peggs KS, Furness AJS, Fisher R, Marafioti T, et al. Ultra-deep T cell receptor sequencing reveals the complexity and intratumour heterogeneity of T cell clones in renal cell carcinomas. *The Journal of pathology*. 2013;231(4):424-32. doi: 10.1002/path.4284. 462
9. Rosati E, Dowds CM, Liaskou E, Henriksen EKK, Karlsen TH, Franke A. Overview of methodologies for T-cell receptor repertoire analysis. *BMC Biotechnology*. 2017;17(1). doi: 10.1186/s12896-017-0379-9. 463
10. Alves Sousa AdP, Johnson KR, Ohayon J, Zhu J, Muraro PA, Jacobson S. Comprehensive Analysis of TCR- β Repertoire in Patients with Neurological Immune-mediated Disorders. *Scientific Reports*. 2019;9(1). doi: 10.1038/s41598-018-36274-7. 464
11. Cui JH, Lin KR, Yuan SH, Jin YB, Chen XP, Su XK, et al. TCR repertoire as a novel indicator for immune monitoring and prognosis assessment of patients with cervical cancer. *Frontiers in Immunology*. 2018;9(NOV). doi: 10.3389/fimmu.2018.02729. 465
12. Sidhom J-W, Larman HB, Pardoll DM, Baras AS. DeepTCR: a deep learning framework for revealing structural concepts within TCR Repertoire. *bioRxiv*. 2018:464107-. doi: 10.1101/464107. 466
13. Farmanbar A, Kneller R, Firouzi S. RNA sequencing identifies clonal structure of T-cell repertoires in patients with adult T-cell leukemia/lymphoma. *npj Genomic Medicine*. 2019;4(1). doi: 10.1038/s41525-019-0084-9. 467
14. Kaplinsky J, Arnaout R. Robust estimates of overall immune-repertoire diversity from high-throughput measurements on samples. *Nature Communications*. 2016;7. doi: 10.1038/ncomms11881. 468
15. Sethna Z, Elhanati Y, Callan CG, Walczak AM, Mora T. OLGA: fast computation of generation probabilities of B- and T-cell receptor amino acid sequences and motifs. *Bioinformatics*. 2019. doi: 10.1093/bioinformatics/btz035. 469
16. Tickotsky-Moskovitz N, Dvorkin S, Rotkopf A, Kuperman AA, Efroni S, Louzoun Y. CDR3 and V- genes show distinct reconstitution patterns in T-cell repertoire post allogeneic bone marrow transplantation. *bioRxiv*. 2020:2020.03.05.977900-2020.03.05. doi: 10.1101/2020.03.05.977900. 470

17. Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science*. 2006;313(5786):504-7. doi: 10.1126/science.1127647. 504
18. Parzen E. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*. 1962;33(3):1065-76. doi: 10.1214/aoms/1177704472. 505
19. Schrama D, Ritter C, Becker JC. T cell receptor repertoire usage in cancer as a surrogate marker for immune responses. *Seminars in immunopathology*. 2017;39(3):255-68. doi: 10.1007/s00281-016-0614-9. 506
20. Hochreiter S, Schmidhuber J. Long Short-Term Memory. *Neural Computation*. 1997;9(8):1735-80. doi: 10.1162/neco.1997.9.8.1735. 507
21. Springer I, Besser H, Tickotsky-Moskovitz N, Dvorkin S, Louzoun Y. Prediction of specific TCR-peptide binding from large dictionaries of TCR-peptide pairs. *Frontiers in immunology*. 2020;11:1803. 508
22. Glanville J, Huang H, Nau A, Hatton O, Wagar LE, Rubelt F, et al. Identifying specificity groups in the T cell receptor repertoire. *Nature*. 2017;547(7661):94-8. doi: 10.1038/nature22976. 509
23. Sidhom J-W, Bessell CA, Havel JJ, Kosmides A, Chan TA, Schneck JP. ImmunoMap: A Bioinformatics Tool for T-cell Repertoire Analysis. *Cancer immunology research*. 2018;6(2):151-62. doi: 10.1158/2326-6066.CIR-17-0114. 510
24. Oakes T, Heather JM, Best K, Byng-Maddick R, Husovsky C, Ismail M, et al. Quantitative Characterization of the T Cell Receptor Repertoire of Naïve and Memory Subsets Using an Integrated Experimental and Computational Pipeline Which Is Robust, Economical, and Versatile. *Frontiers in immunology*. 2017;8:1267-. doi: 10.3389/fimmu.2017.01267. 511
25. Pogorelyy MV, Minervina AA, Touzel MP, Sycheva AL, Komech EA, Kovalenko EI, et al. Precise tracking of vaccine-responding T cell clones reveals convergent and personalized response in identical twins. *Proceedings of the National Academy of Sciences of the United States of America*. 2018;115(50):12704-9. doi: 10.1073/pnas.1809642115. 512
26. Toledano A, Elhanati Y, Benichou JIC, Walczak AM, Mora T, Louzoun Y. Evidence for Shaping of Light Chain Repertoire by Structural Selection. *Frontiers in Immunology*. 2018;9:1307-. doi: 10.3389/fimmu.2018.01307. 513
27. Pickman Y, Dunn-Walters D, Mehr R. BCR CDR3 length distributions differ between blood and spleen and between old and young patients, and TCR distributions can be used to detect myelodysplastic syndrome. *Physical biology*. 2013;10(5):056001-. doi: 10.1088/1478-3975/10/5/056001. 514
28. Elhanati Y, Sethna Z, Callan CG, Mora T, Walczak AM. Predicting the spectrum of TCR repertoire sharing with a data-driven model of recombination. *Immunological reviews*. 2018;284(1):167-79. doi: 10.1111/imr.12665. 515
29. Postow MA, Manuel M, Wong P, Yuan J, Dong Z, Liu C, et al. Peripheral T cell receptor diversity is associated with clinical outcomes following ipilimumab treatment in metastatic melanoma. *Journal for ImmunoTherapy of Cancer*. 2015;3(1). doi: 10.1186/s40425-015-0070-4. 516
30. Rudqvist N-P, Pilones KA, Lhuillier C, Wennerberg E, Sidhom J-W, Emerson RO, et al. Radiotherapy and CTLA-4 Blockade Shape the TCR Repertoire of Tumor-Infiltrating T Cells. *Cancer immunology research*. 2018;6(2):139-50. doi: 10.1158/2326-6066.CIR-17-0134. 517
31. Dvorkin S. Autoencoder/cancer data IEDB at master · shirtdvir/Autoencoder · GitHub. 2020. 518
32. Kullback S, Leibler RA. On Information and Sufficiency. *The Annals of Mathematical Statistics*. 1951;22(1):79-86. doi: 10.1214/aoms/1177729694. 519
33. Silverman BW. *Density estimation for statistics and data analysis*: CRC press; 1986. 520
34. Sawyer SF. *Analysis of Variance: The Fundamental Concepts*. *Journal of Manual & Manipulative Therapy*. 2009;17(2):27E-38E. doi: 10.1179/jmt.2009.17.2.27e. 521

35.	Johnson SC. Hierarchical clustering schemes. Psychometrika. 1967;32(3):241-54. doi:	554
	10.1007/BF02289588.	555
36.	Maaten Lvd, Hinton G. Visualizing data using t-SNE. Journal of machine learning	556
	research. 2008;9(Nov):2579-605.	557
37.	Dvorkin S, Levi R, Louzoun Y. Autoencoder based local T cell repertoire density can be	558
	used to classify samples and T cell receptors. bioRxiv. 2020:2020.06.12.148502. doi:	559
	10.1101/2020.06.12.148502.	560
		561
		562

Figure captions

563

Fig. 1. A. Autoencoder description. The autoencoder is composed of two models: encoder and decoder. Encoder's input: a vector of n dimensions of concatenated one-hot (oh) vectors, representing a CDR3 sequence. Encoder's output (decoder's input): the embedded representation with 30 dimensions. Decoder's output: the reconstructed vector. The autoencoder is trained by minimizing the reconstruction MSE (Mean Squared Error). The classifier is a combination of the encoder (with fixed weights) and 4 more fully connected layers. The classifier is trained by minimizing the BCE loss (Binary Cross Entropy). **B. Data preprocessing and distance maintaining.** Encoder's input: Two $1 \times n$ one-hot representations of TCR sequences (x_1, x_2). Each amino acid is represented by a 21 long one hot representation. The last value is for the stop signal. Each sequence ends with a stop signal. The encoder's output and decoder's input is a 1×30 embedded vector with lower dimension z for each input sequence (z_1, z_2). Decoder's output: a $1 \times n$ reconstructed one-hot vector for each input (x'_1, x'_2). In the EMB model, an additional loss term is added to reduce the difference between the original distances (Euclidean distances between every x_i and x_j) and embedded vectors distances (euclidean distances between every z_i and z_j). The distance is computed on the output vector. However, when reporting the results, the definition of the accuracy is based on a vector after the max argument on each 21 positions is performed.

582

Fig 2. A. Edit distances distribution. Histogram of edit distances between all pairs of CDR3 sequences from two samples. Few sequences have 2 or fewer mismatches. **B. Models glossary.** The table describes the model and model acronym used in all the following figures and in the text. **C. E accuracy.** All accuracies of E for the different datasets with no mismatches (fully matched, red), 1 mismatch (orange), and 2 mismatches (blue). **D. VE accuracy.** All accuracies of VE for the different datasets (except for datasets without Vs). **E. Encoder and CDR3 distances correlation.** Pairwise Euclidean distances in the set of CDR3 one-hot vectors (Xs) in the x axis and pairwise Euclidean distances in the embedded set (Zs) predicted by E in the y axis. The legend contains the Spearman correlation between the axes. **F. Encoder's distances correlation when combined with distances.** Pairwise Euclidean distances in the set of CDR3 one-hot vectors (Xs) in the x axis and pairwise Euclidean distances in the embedded set (Zs) predicted by EM in the y axis. **G. EM accuracy.** All accuracies of EM for the different datasets. **H. VEM accuracy.** All accuracies of VEM for the different datasets.

598

Fig. 3. A. Encoder's projection TSNE (Benny Chain dataset). TCRs with cysteine are colored in red. **B. Encoder's projection TSNE (Benny Chain dataset).** Colored by CDR3 length. **C. Radius vs Publicity (Benny Chain dataset).** The x axis is the number of samples containing this TCR clone and the y axis is the distance from the center of all projections. The Spearman correlation is in the legend. **D. Self-density vs Publicity (Benny Chain dataset).** Axes: y- The KDE between a TCR clone and its sample, x- number of samples that contain this TCR clone. **E. Encoder's projection TSNE (GLIPH).** TCRs with specificity to TRPVTGGGAM peptide are colored in red. The other groups are colored in grey. **F. Encoder's projection TSNE (Cancer dataset).** TCRs from hosts with cancer are colored in red and TCRs from samples collected from

healthy hosts are colored in grey. **G.** Average self-density per feature. The average self-density of all 7 peptides in GLIPH data set and for TCRs with a specific number of G (Glycine), C (Cysteine) and P (Proline). The p-value of t-test is shown as * (<0.05), ** (<0.01), *** (< 0.001).

Fig. 4 A. KDE dendrogram. Hierarchical clustering distances computed on the KDE distance matrix (without the clone frequencies) of the vectors samples predicted by the autoencoder combined with Vs (VE). Samples are colored by hosts and twins are colored is similar colors (P1 and P2: blue and light blue, S1 and S2: red and salmon, Q1 and Q2: spring green and lawn green). **B.** KL dendrogram. Hierarchical clustering distances computed on the KL distance matrix. **C.** Clusters entropies by hosts. All hosts (P1, P2, S1, S2, Q1 and Q2) entropies through the different methods (labeled with a combination of: E= AE, EM=AE+DIS, V= with v genes, F= with frequencies). **D.** Clusters entropies by time-points. All time-points (7 D pre, 0 D post, 7 D post, 15 D post, 45 D post and 2 Y post-vaccination) entropies through the different methods.

Fig. 5. Encoders' self-density (Benny Chain dataset). **A.** Average KDE density within the set of vectors predicted by the basic autoencoder (E). **B.** Average KDE density within the set of vectors predicted by the basic autoencoder combined with the clone frequencies (EF). **C.** Average KDE density within the set of vectors predicted by the autoencoder combined with the CDR3 distances (EM). **D.** Average KDE density within the set of vectors predicted by the autoencoder combined with the CDR3 distances and the clone frequencies (EMF).

Tables

633

Table 1. Description of each category.

634

DATA SET	TAGS	Description
GLIPH	CTELKLSDY	TCRs with shared specificity to CTELKLSDY peptide
	VTEHDTLLY	TCRs with shared specificity to VTEHDTLLY peptide
	GLCTLVAML	TCRs with shared specificity to GLCTLVAML peptide
	GILGFVFTL	TCRs with shared specificity to GILGFVFTL peptide
	TPRVTGGGAM	TCRs with shared specificity to TPRVTGGGAM peptide
	NLVPMVATV	TCRs with shared specificity to NLVPMVATV peptide
	LPRRSGAAGA	TCRs with shared specificity to LPRRSGAAGA peptide
ImmunoMap	TRP2	TCRs responding to TRP2 immunotherapy in tumor-bearing mice
	SIY	TCRs responding to SIY immunotherapy in tumor-bearing mice
Benny Chain	Memory	TCR samples from memory cells
	Naive	TCR samples from naïve cells
Vaccine	7 D Pre	TCR samples collected 7 days before immunization
	0 D Post	TCR samples collected in the day of immunization
	7 D Post	TCR samples collected 7 days after immunization
	15 D Post	TCR samples collected 15 days after immunization
	45 D Post	TCR samples collected 45 days after immunization
	2 Y Post	TCR samples collected 2 years after immunization
Cancer	Healthy	TCRs collected from healthy hosts
	Cancer	TCRs collected from hosts with different types of cancers

635

Table 2. AUC values for each category for our four models (*E*, *VE*, *EM*, *VEM*) and

636

DeepTCR.

637

DATASET	CATEGORY	E AUC	VE AUC	EM AUC	VEM AUC	Deep-TCR
GLIPH	CTELKLSDY	0.4	x	0.47	x	0.5
	VTEHDTLLY	0.58		0.6		0.7
	GLCTLVAML	0.64		0.64		0.6
	GILGFVFTL	0.61		0.67		0.65
	TPRVTGGGAM	0.81		0.54		0.83
	NLVPMVATV	0.65		0.56		0.73
	LPRRSGAAGA	0.58		0.47		0.58
	Average	0.61		0.56		0.65
ImmunoMap	TRP2	0.78	0.74	0.74	0.8	0.71

	SIY	0.68	0.79	0.78	0.82	0.71
	Average	0.73	0.765	0.76	0.81	0.71
Benny Chain	Memory	0.58	x	0.57	x	x
	Naive	0.57		0.57		
	Average	0.575		0.57		
Vaccine	15 D Post	0.54	0.53	0.52	0.52	x
	7 D Post	0.5	0.49	0.48	0.48	
	2 Y Post	0.6	0.63	0.64	0.62	
	0 D Post	0.49	0.49	0.49	0.48	
	7 D Pre	0.51	0.51	0.49	0.5	
	45 D Post	0.49	0.51	0.5	0.48	
	Average	0.52	0.52	0.52	0.51	
Cancer	Healthy	0.87	0.91	0.93	0.91	x
	Cancer	0.94	0.94	0.86	0.91	
	Average	0.9	0.925	0.895	0.91	

638

Table 3. The table contains the dataset name, the categories in each dataset, the number of sequenced libraries for each category, the number of clones in each category, whether the dataset contains V gene usage, and whether the dataset details the clone frequencies.

Dataset	Category	Samples	Clones	Number of different V genes	Frequencies
GLIPH	CTELKLSDY	1	25	X	X
	VTEHDTLLY	1	271		
	GLCTLVAML	1	739		
	GILGFVFTL	1	586		
	TPRVTGGGAM	1	65		
	NLVPMVATV	1	166		
	LPRRSGAAGA	1	214		
ImmunoMap	TRP2	3	488	36	V
	SIY	4	1,106		
TILs Immunotherapies	9H10	5	3,643	27	V
	RT	5	6,707		
	9H10+RT	5	7,826		
	Control	5	3,643		

Benny Chain	Memory	36	675,146	X	V
	Naive	20	1,557,365		
Vaccine	7 D Pre	17	16,786	56	V
	0 D Post	28	27,781		
	7 D Post	24	23,700		
	15 D Post	23	22,320		
	45 D Post	30	29,646		
	2 Y Post	3	2,260		
Cancer	Healthy	1	7,342	82	V
	Cancer	1	409		

643

Table 4. Glossary of terms used for the analysis.

644

l	Length of CDR3 sequence
m	Maximal length of CDR3 sequence in an analyzed repertoire
n	Dimension of one-hot representation
n_v	Number of possible Vs in a repertoire
N	Size of the training set
b	Batch size
D_N	Distance matrix of the all Euclidean distances in the training set
D_b	Distance matrix of the all Euclidean distances in a batch
x	Encoder's input vector
z	Encoder's output vector- embedded vector
Z	Length of z
s	The sequence form of x
x'	Reconstructed vector from the decoder's output vector-
s'	The sequence form of x'
V	Set of all different V genes
R	Number of randomly sampled clones
E	R by R edit distance matrix

645

A

B

B

	Description
E	Encoder with MSE loss
EM	Embedding with MSE loss + DIS loss
VE	Encoder with MSE loss + V
VEM	Embedding with MSE loss + DIS loss + V

A

E Self-density by category

B

EF Self-density by category

C

EM Self-density by category

D

EMF Self-density by category