CS-E4880 - Machine Learning in Bioinformatics
Determining epitope-specificity of TCRs with Gaussian processes
Advance Access Publication Date: Day Month Year
Manuscript Category

Aalto-yliopisto

Course Projects

Determining epitope-specificity of TCRs with Gaussian processes

Santeri Mentu [1,*], Esa Turkulainen [1,*]

[1] Department of Computer Science, Aalto University, Espoo, 02150, Finland

[*] To whom correspondence should be addressed.

Associate Editor: Emmi Jokinen

Received on 15.5.2019; revised on 29.5.2019; accepted on XXXXX

Abstract

Motivation: T cells are immune system cells that are primarily responsible for the detection and elimination of cells that are infected with viruses or other antigens. In order to mount an immune response, T cell receptors must recognize the foreign antigen being presented by infected cells. This mechanism determines much of the cell-mediated adaptive immune response, and is therefore of interest in the diagnosis and treatment of various diseases and disorders. We aim to model and predict this specificity by sequencing the amino acid sequences making up the TCR and using a machine learning technique called Gaussian processes to predict epitope specificity.

Results: We were able to achieve a mean LOSO AUC scores between 0.74 and 0.80 by using a k-mer based approach with automatic relevance determination (ARD). The performance of our method is predictably worse than that of alignment-based methods, which better incorporate domain knowledge into the feature representation.

Availability: All datasets used are publicly available from https://vdjdb.cdr3.net/.

Contact: santeri.mentu@aalto.fi, esa.turkulainen@aalto.fi

1 Introduction

T cells and B cells are the lymphocytes that form the basis for the function of the adaptive immune system. B cells are responsible for humoral, antibody-driven adaptive immunity and T cells are responsible for cell-mediated, cytoxic adaptive immunity. In order for T cells to mount an immune response, they must recognize a major histocompatibility complex (MHC) with an antigen in the form of a peptide bound to it, forming a pMHC. The binding happens through the T cell receptor (TCR) on the surface of the T cell. These receptors have relatively low affinity and are degenerate, meaning that multiple TCRs can recognize the same pMHC, but individual TCRs generally only bind to a limited set of pMHCs (Alberts *et al.* 2002).

TCRs are primarily formed from alpha and beta chain, but there are also TCRs made up of gamma and sigma chains. Three complementary determining regions (CDRs) define the way the TCR interacts with the peptide and its binding groove, with CDR3 interacting most strongly with the peptide and the other two more with the binding groove (Jokinen *et al.* 2019). The binding of a TCR to an antigen peptide bound to a MHC is shown in figure 1. The challenges of predicting epitope specificity primarily stem from the complexity and magnitude of the possible interactions. It is estimated that the process by which T cells mature can result in 10^{18} different TCRs (Robins *et al.* 2009), and that the TCR repertoire of each individual contains an approximate 10^6 unique TCR sequences. It has been estimated that each of these TCRs can interact with at least 1 million pMHCs (cross-reactivity), and a given pMHC can potentially elicit responses from millions of TCRs

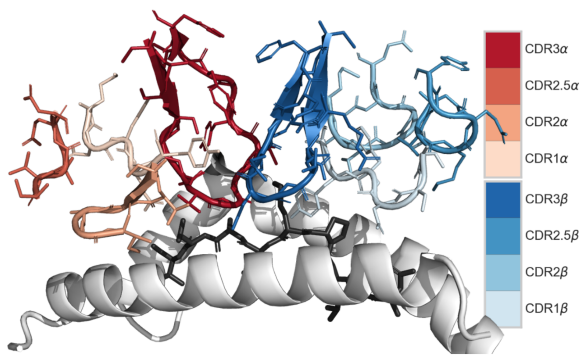


Fig. 1. Rendering of the TCR binding to a MHC and antigen peptide by (Jokinen et al. 2019).

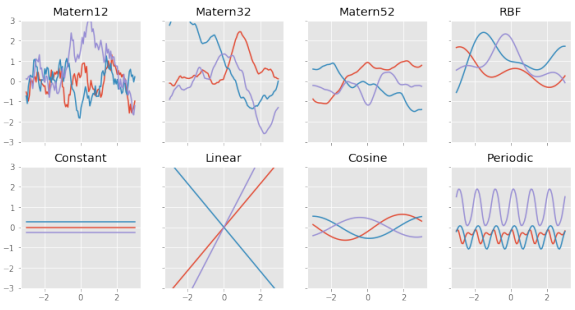


Fig. 2. Examples of various kernels from the documentation of GPFlow by (Matthews et al. 2017).

(Woolridge *et al.* 2012). This means that modelling and predicting all possible pMHC-TCR interactions using numerical simulations would be impossible.

2 Approach

2.1 Gaussian Processes

The difficulty of predicting epitope specificities exactly has led to several machine learning approaches being developed. These include methods based on clustering, random forests based on biophysical features, and kernel methods. An enigmatic property of TCR specificity is that TCRs are simultaneously specific and cross-reactive (Nishant *et al.* 2017). This also poses a challenge to when trying to predict specificity using simple models. One of the most successful machine learning techniques to have been applied to this challenge are Gaussian processes. Gaussian processes have several qualities well suited for this task, including being probabilistic, robust even for small datasets, and scaling well to larger datasets that may become available in the future.

Gaussian processes define a prior over functions which can be used for Bayesian regression (Snelson). A Gaussian process is a continuous stochastic process, where any finite set of function variables $\mathbf{f} = \{f_1, f_2, \ldots, f_n\}$ follows a multivariate Gaussian distribution with some covariance matrix $\mathbf{K}$

$$p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \mathbf{K}).$$

This covariance matrix is constructed using a covariance function, or kernel, which gives a measure of similarity between points. Different kernels result in different properties to the family of functions defined by the Gaussian process. Figure 2 shows examples of GPs with different kernel functions. We experimented with using a radial basis function (RBF) or squared exponential kernel, a rational quadratic kernel, and Matérn kernels.

The RBF kernel is defined as

$$k(\mathbf{x}, \mathbf{x}'|\theta) = \sigma^2 \left(-\frac{(\mathbf{x} - \mathbf{x}')^T(\mathbf{x} - \mathbf{x}')}{2l^2} \right),$$

where σ is the magnitude hyperparameter and l is the lengthscale hyperparameter. A high lengthscale means that the variable has weak predictive power. The rational quadratic kernel is defined as

$$k_{\text{RQ}}(r) = \left(1 + \frac{r^2}{2\alpha l^2} \right)^{-\alpha},$$

where r is the euclidean distance between the inputs, and l and α are hyperparameters. The Matérn kernel is defined as

$$k_{\text{Matérn}}(r) = \frac{2^{1-v}}{\Gamma(v)} \left(\frac{\sqrt{2v}r}{l} \right)^v K_v \left(\frac{\sqrt{2v}r}{l} \right),$$

where v and l are hyperparameters. The parameter v takes values $v = p+1/2, p \in \mathbb{N}^+$, so the Matérn kernels are commonly named `Matern12` for $v = 1/2$, `Matern32` for $v = 3/2$, and so on. The different kernels all have the effect of making samples close to each other in the feature space have similar epitope specificities, but different kernels result in different shapes for the fall-off.

Training the models is done by optimizing the model hyperparameters such that we minimize the marginal likelihood

$$p(\mathbf{y}|\theta) = \int p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|\theta) \mathrm{d}\,\mathbf{f},$$

where θ are model parameters, such as lengthscales of the kernel.

Gaussian processes are an example of nonparametric lazy learning methods, so the model is only evaluated during posterior prediction. However, this has complexity $\mathcal{O}(n^3)$, so we use some sort of approximation such as Laplace approximations, or in our case, sparse variational GPs. Sparse variational methods employ introduce M fiducial points at optimized locations of the feature space, which results in a computational complexity of $\mathcal{O}(NM^2)$

2.2 Feature representation

Since the Gaussian processes operate on data of fixed dimensionality, the variable length TCR sequences have to be transformed into a suitable feature representation. One highly successful approach is to align the sequences by introducing a gap in the middle and constructing the covariance function using a substitution matrix such as BLOSUM62. This is the approach taken by Jokinen *et al.* (2019). We chose to use a feature representation based on k-mers.

3 Methods

3.1 Data

The data is sequenced human TCR data corresponding to four different antigen epitopes. For the training, the data also contains TCR sequences chosen at random, which are assumed to not recognize the epitope. All data is publicly available from VDJdb.

3.2 Leave-one-subject-out cross-validation

We used leave-one-subject-out (LOSO) cross validation, where we leave out all samples from one subject and train the model on the rest. We then validate the model by predicting the epitope specificities for the excluded subject. This is also the method used by (Jokinen *et al.* 2019), since it is quite representative of the the actual usage scenario for the method. We cross-validate using subjects with more than 10 samples. For each subject we chose a set of control samples that was removed from the training set and used in the cross-validation. The number of control samples was equal to the number of samples for that subject.

3.3 GPflow

We used the software library GPflow (Matthews *et al.* 2017) for our experiments. This library is built on top of the Tensorflow framework to provide GPU-accelerated training of Gaussian processes. The training algorithm also performs automatic relevance determination (ARD), which optimizes the lengthscales of different dimensions separately.

3.4 Feature vectors

Feature vectors for the TCRs were constructed by picking a set of k-mers present in the samples, and then encoding the presence of these k-mers in each sequence to a binary vector. This feature presentation

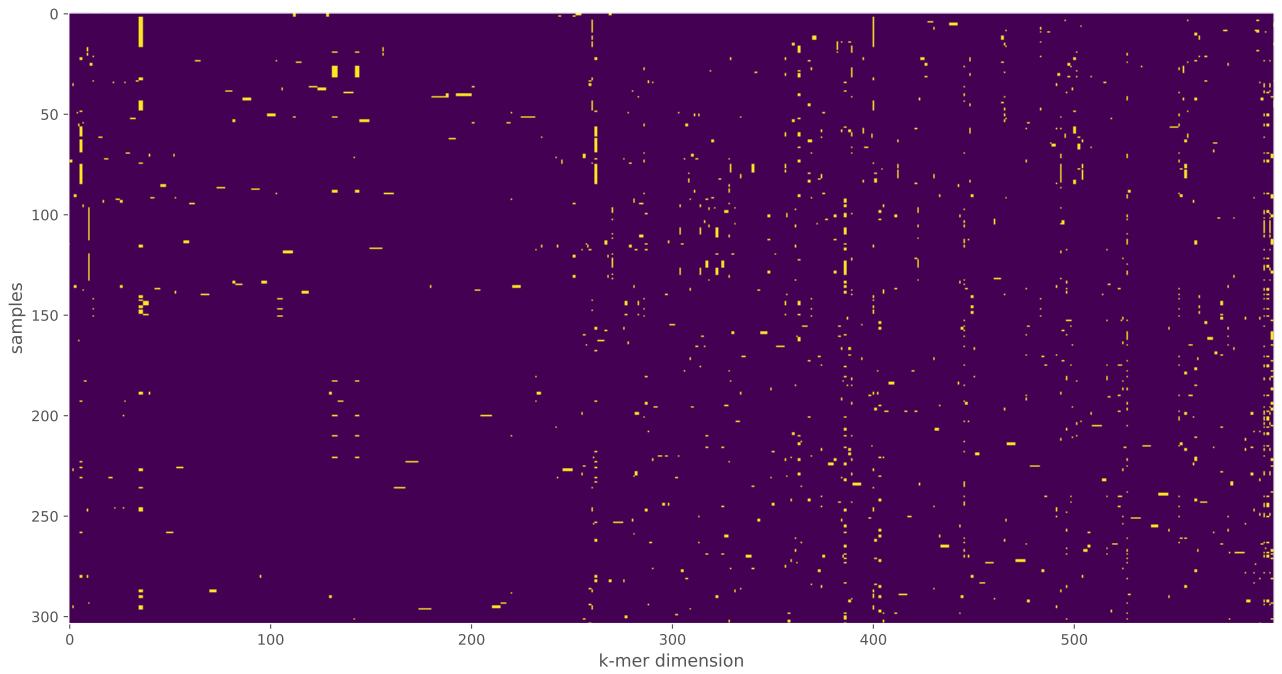


Fig. 3. Visualization of our feature vectors from the ATDALMTGY data. The top half contains feature vectors for the samples and the bottom half corresponds to the control samples. The dimensions are sorted in order of increasing lengthscale hyperparameter, meaning that the dimensions on the left are determined to be more important to the regression.

is visualized in figure 3. We chose the set of k-mers used by training the model on all k-mers and using all TCRs and then picking k-mers in order of increasing lengthscale from the ARD. We selected to use 3-mers to construct feature vectors, since they are the most commonly used in high-throughput bioinformatics. We used CDR3βsequences associated with four different epitopes.

We also experimented with a model combining both 3-mers and 4-mers in the feature representation. We performed the lenghtscale feature selection process independently for the 3-mers and 4-mers and then constructed a the kernel as the sum of two rational quadratic kernels, corresponding to the two feature representations.

3.5 Training method

By using sparse variational Gaussian processes, it is possible to train GP models that scale favorably to large and high-dimensional datasets. In order to model specificities in the form of a probability, our model combines a Gaussian prior with a probit link function.

4 Results

In order to optimize the model we performed a hyperparameter grid search using different kernels and dimensionalities of the feature representation. We used the dataset `vdj_human_ATDALMTGY` for each of the hyperparameter search runs, since multiple subjects with a usable number of samples. The best accuracy of 0.785 was achieved with 600 dimensions and the Rational Quadratic kernel. All of the kernels performed the best with 600 dimensions (maximum number tested) and the Rational Quadratic performed best through all tested numbers of dimensions. The differences may not be significant between the kernels however, as the results might vary between runs more than the largest difference between best performers (0.015).

The best feature selection method across all kernels and dimension sizes was Automatic Relevance Detection. The search for largest relative or absolute differences between k-mer frequencies in the training and control sets, though intuitive, proved insufficient in comparison to using lengthscales derived from ARD.

Table 1. Results of grid search

kernel	200	400	600
RBF	0.677	0.704	0.750
Matern52	0.678	0.709	0.758
Matern32	0.677	0.708	0.763
Matern12	0.682	0.716	0.771
RQ	0.686	0.731	**0.785**

Table 2. Method performance

epitope	3-mer	hybrid
vdj human ATDALMTGY	0.785	0.802
vdj human TPRVTGGGAM	0.743	0.712
vdj human RAKFKQLL	0.799	0.834
vdj human NLVPMVATV	0.684	0.706

All results from our grid search have been compiled to Table 1. The receiver operating characteristic curve for all of the subjects using the RQ kernel and 600 dimensions is shown in Figure 4.

The combination kernel performed slightly better on average. In testing we used 600 dimensions for the 3-mer kernel and 600 for the 4-mer kernel, although the number of unique 4-mers present in the data is larger than the number of 3-mers and therefore it might make sense to use higher dimensionalities for the corresponding kernel.

We trained the top performing model and the hybrid model on four different epitope datasets. The results are presented in table 2 and the AUC plots for the 3-mer model are presented in appendix A figure 5. The 3-mer model has an accuracy between 0.74 and 0.8, and the hybrid model performed on average slightly better with accuracies between 0.74 and 0.83.

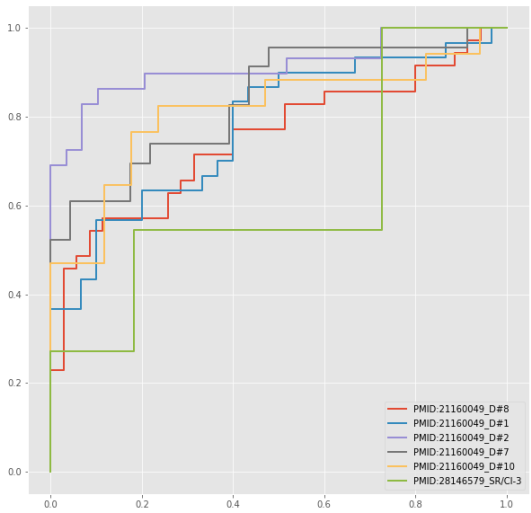


Fig. 4. AUC curves for different subjects for the Rational Quadratic kernel on different subjects using 600 dimensions.

5 Discussion

In this project our primary goal was to find the best 3-mer feature selection method, the best Gaussian process kernel out of two simple kernels and the optimal number of dimensions to be used with those kernels, for an epitope-specificity determination task. We also experimented briefly with using two kernels in combination, with some success. We chose data compiled for a previously conducted study by (Jokinen *et al.* 2019) so that our results would be directly comparable to the results achieved in that study.

Our results showed that, in the case of the 3-mer models, the Rational Quadratic kernel performed the best across all numbers of dimensions, and that the model performance kept increasing with the number of dimensions used. Due to computational complexity and time limitations, the highest number of dimensions tried was 600. It is also worth noting that the accuracies of the models vary considerably between subjects and epitopes. This is caused by the fact that subjects vary in their TCR repertoires and in what and how many TCRs were sampled from each subject for the dataset. This has the effect of making model performance rather varied, which is less than ideal for any medical applications. Regardless, the best performing model was Rational Quadratic between all subjects in our tests.

Finally, we know from other research into the topic that more advanced feature representations are capable of superior performance. Other, perhaps task-specific kernels may also yield performance improvements. Our experiments with multiple kernel learning did also give initially promising results.

6 Conclusion

Applying automatic relevance determination on a 3-mer feature space proves to be a somewhat effective method for selecting features for determining epitope-specificities by Gaussian processes, with a variety of kernels giving very similar results. While our approach did not achieve the accuracies of the current state-of-the-art methods, it still remains as an interesting avenue for future research into feature representation for Gaussian processes in this field.

References

[Alberts *et al.* 2002]Alberts B, Johnson A, Lewis J, et al. *Molecular Biology of the Cell*. 4th edition. New York: Garland Science; 2002. Chapter 24, The Adaptive Immune System. Available from: https://www.ncbi.nlm.nih.gov/books/NBK21070/

[Jokinen *et al.* 2019]Emmi Jokinen, Markus Heinonen, Jani Huuhtanen, Satu Mustjoki and Harri Lähdesmäki (2019) TCRGP: Determining epitope specificity of T cell receptors

[Robins *et al.* 2009]Robins HS, et al. Comprehensive assessment of T-cell receptor β-chain diversity in $\alpha\beta$ T cells. *Blood.* 2009;114:4099âĽ“4107. doi: 10.1182/blood-2009-04-217604.

[Woolridge *et al.* 2012]Wooldridge, L. et al. (2012). A single autoimmune T cell receptor recognizes more than a million different peptides. *Journal of Biological Chemistry*, 287(2), 1168-1177.

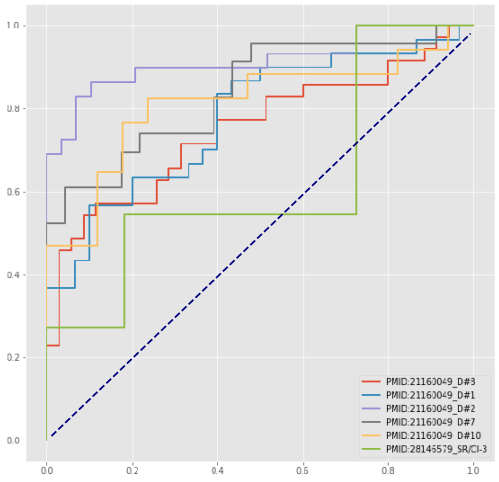
[Nishant *et al.* 2017]Nishant K. Singh, Timothy P. Riley, Sarah Catherine B. Baker, Tyler Borrman, Zhiping Weng, Brian M. Baker. Emerging Concepts in TCR Specificity: Rationalizing and (Maybe) Predicting Outcomes. *The Journal of Immunology* October 1, 2017, 199 (7) 2203-2213; DOI: 10.4049jimmunol.1700744

[Snelson]Cambridge University Department of Engineering Advanced Lecture Series, Lecture 6: Gaussian Process Models for Machine Learning, Cambridge University, viewed 15 May 2019, <http://mlg.eng.cam.ac.uk/tutorials/06/es.pdf/>.

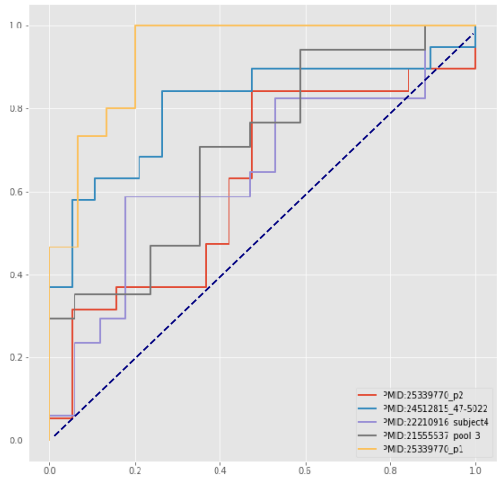
[Matthews *et al.* 2017]Alexander G. de G. Matthews, Mark van der Wilk, Tom Nickson, Keisuke Fujii, Alexis Boukouvalas, Pablo Leòn-Villagrà, Zoubin Ghahramani, James Hensman. GPflow: A Gaussian Process Library using TensorFlow. *Journal of Machine Learning Research* 18 (2017) 1-6.

A AUC plots

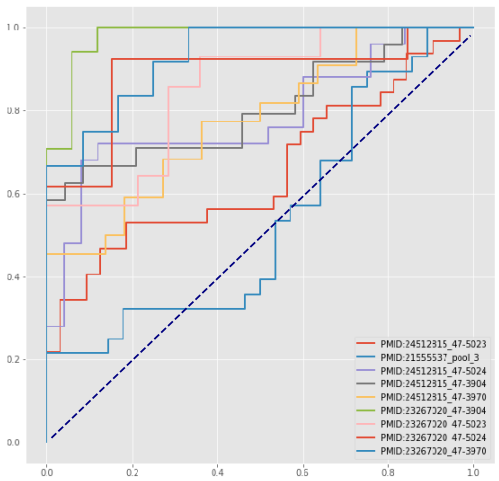
ATDALMTGY



TPRVTGGGAM



RAKFKQLL



NLVPMVATV

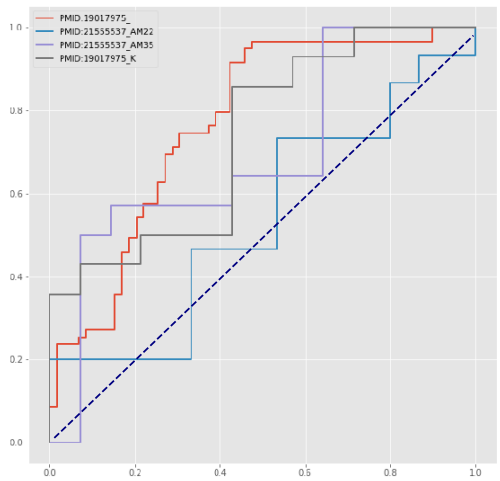


Fig. 5. AUC plots for different epitopes using Rational Quadratic kernel and 600 dimensions.