

# Using Gaussian process with test rejection to detect T-cell epitopes in pathogen genomes

Liwen You, Vladimir Brusic, Marcus Gallagher and Mikael Bodén

**Abstract**—A major challenge in the development of peptide-based vaccines is finding the right immunogenic element, with efficient and long-lasting immunisation effects, from large potential targets encoded by pathogen genomes. Computer models are convenient tools for scanning pathogen genomes to pre-select candidate immunogenic peptides for experimental validation. Current methods predict many false positives resulting from a low prevalence of true positives.

We develop a test reject method based on the prediction uncertainty estimates determined by Gaussian process regression. This method filters false positives amongst predicted epitopes from a pathogen genome. The performance of stand-alone Gaussian process regression is compared to other state-of-the-art methods using cross-validation on eleven benchmark data sets. The results show that the Gaussian process method has the same accuracy as the top performing algorithms. The combination of Gaussian process regression with the proposed test reject method is used to detect true epitopes from the Vaccinia virus genome. The test rejection increases the prediction accuracy by reducing the number of false positives without sacrificing the method's sensitivity. We show that the Gaussian process in combination with test rejection is an effective method for prediction of T-cell epitopes in large and diverse pathogen genomes, where false positives are of concern.

**Index Terms**—Immunology, amino acid sequence, epitope, machine learning, Gaussian processes, regression, false positives.

## I. INTRODUCTION

CD8+ T-cells recognize peptides presented by major histocompatibility complex (MHC) class I molecules on the surface of cells. This recognition is critical for identification of infected or cancer cells and their subsequent destruction by the immune system. The antigenic peptide presentation involves primarily three steps: proteasomal cleavage, transporter associated with antigen processing (TAP) and MHC-peptide binding. The MHC-peptide binding is by far the most discriminant step. Only a minority of peptides will bind MHC molecules (1-5%) while the number of peptides that elicit immune responses, known as T-cell epitopes, is even lower [1]. Accurate prediction of MHC-peptide binding affinity is thus essential for identification of immunologically relevant peptides for use in vaccine formulations.

Many different attempts have been made to predict MHC peptide binding. There are primarily three approaches: a structural approach, a sequence-based approach and a hybrid structural/sequence-based approach [2]. Sequence-based approaches can be further categorized into motif/profile based methods (including RankPep [3], Average Relative Binding (ARB) [4], Stabilized Matrix (SMM) [5], [6], Udaka [7] and Parker [8] matrices) and machine learning methods (artificial neural networks (ANN) [9], [10], support vector machines (SVM) [11]–[14], hidden Markov models [15] and DistBoost [16]). These methods are developed and evaluated using different data sets,

which makes it intrinsically difficult to compare their performance. However, it appears that no single approach dominates all the others in terms of accuracy of predictions.

MHC-peptide binding prediction tools can be used to scan the whole genomes of pathogens to identify potential immunogenic peptides for peptide-based vaccine design. Recently, Moutaftsi *et al.* used a consensus approach based on four methods (ARB, SMM, Parker and Udaka) to score H-2<sup>b</sup>-restricted CD8+ T-cell epitopes in the whole genome of Vaccinia virus (VACV) after infection of mice [17]. Their consensus approach is to calculate the median values of four ranks from the four methods on all possible octamers, nonamers and decamers from the VACV genome for the mouse H-2 molecules K<sup>b</sup> and D<sup>b</sup>. The final ranks of peptides indicate relative epitope probabilities. The top 1% of peptides within each size-based class were selected to test for antigenicity *in vitro*. Out of these several thousand candidates, only 49 peptides were identified as positive samples. Therewith, these 49 peptides were tested for their capacities to bind purified K<sup>b</sup> and D<sup>b</sup> molecules *in vitro*. All 49 epitopes were shown to be true MHC binders, some with very high binding affinities, illustrating the utility of scoring methods for pre-selection of candidate T-cell epitopes.

Prediction methods have the potential to simplify experimental design and effort by reducing the number of targets that need to be screened. By treating strength of prediction as an indicator of confidence of prediction, Moutaftsi *et al.* were able to isolate a small group of candidate epitopes. A large number of unpromising targets could be eliminated from a prohibitively long list. Most likely, this elimination process also discarded some of the good targets. This illustrates a growing need for improved computational methods to provide more accurate simulations of biological processes. When combined with experimental methods, such methods can further improve outcomes of vaccine research while lowering the cost.

Gaussian processes (GP) have been used successfully in many different machine learning studies [18], [19], but their application in bioinformatics are still at early stages. In this study, we explore the use of Gaussian processes to the MHC-binding problem. More specifically, we investigate the use of the Gaussian process sample prediction variance as a natural indicator of prediction confidence. We first estimate the accuracy of Gaussian process models applied to MHC-peptide binding prediction and compare their performance to other state-of-the-art methods. We then extend the method by employing prediction confidence levels to reduce the number of false positives and identify a small number of correctly predicted experimental targets. The method is evaluated on the full VACV genome, which represents a challenging problem with a large and realistically diverse input space. We also validate the method's ability to reduce false positive predictions on peptides for MHC class I alleles in human, mouse, chimp and macaque.

## II. METHODS

### A. Data representation

The MHC-binding affinity data set consists of short amino acid sequences. We used the conventional orthogonal coding to represent peptides, where each amino acid of 20 standard amino acids is coded as 20 bit binary vectors, where a single position contains one and the others are zeros. For example, an eight amino acid residues long peptide is coded as a vector with  $20 \times 8 = 160$  dimensions.

### B. Gaussian processes

Gaussian processes provide an approach to data modelling that has received increased attention in the machine learning community in recent years [19]. A Gaussian process is a particular stochastic process which can be used to describe a distribution over the properties of functions. Given a set of training data, Gaussian processes provide an effective framework for Bayesian learning. A prior  $p(y(x))$  is specified directly over function space, instead of parameterizing  $y(x)$ , and the training set is used to transform this prior into a posterior of unknown functions. Gaussian processes are an example of the class of techniques known as kernel machines (including Support Vector Machines), but are distinguished by their probabilistic approach to data modelling.

A Gaussian process is fully specified by its mean and covariance functions. The mean function is typically assumed to be zero  $\mu(x) = E(y(x)) = 0$  (assuming centred training data), while the covariance function  $C(x, x')$ , describes the covariance between the values of function  $y$  at any two data points  $x$  and  $x'$ . For any set of variables  $Y(x_1), \dots, Y(x_n)$ , a Gaussian process implies a joint multivariate Gaussian distribution.

The Gaussian process can be used for both classification and regression purposes. Since our problem is to predict MHC-peptide binding affinity, the discussion is confined to Gaussian process regression.

For a regression problem, we want to predict the function value  $y(x_*)$  over the new input  $x_*$ , given a set of input points  $x^n = x_1, x_2, \dots, x_n$  and measured values  $t = t_1, \dots, t_n$ . We assume that the measured values are obtained from the corresponding function value  $y_i$  with additive noise,  $t_i = y_i + \epsilon_i$ , where  $i = 1, \dots, n$  and  $\epsilon$  is an independent zero-mean Gaussian random variable with variance  $\sigma_\nu^2$ . The Gaussian process specifies a prior distribution over  $y_i$  is given by  $\mathbf{Y} \sim N(\mathbf{0}, K)$ , where  $K$  is the  $n \times n$  (non-negative definite) covariance matrix with elements  $K_{ij} = C(x_i, x_j)$ . The prior distribution of the targets is  $N(\mathbf{0}, K + \sigma_\nu^2 I_n)$ , where  $I_n$  is the  $n \times n$  identity matrix. To predict the function values  $y_*$  on the new point  $x_*$ , we are interested in the predictive distribution  $P(y_*|t)$ . Since  $P(t, y_*)$  and  $P(y_*)$  have Gaussian distributions,  $P(y_*|t)$  is also Gaussian and can be calculated in closed form, with mean

$$\hat{y}(x_*) = \mathbf{k}^T(x_*)(K + \sigma_\nu^2 I_n)^{-1} \mathbf{t} = \sum_{i=1}^n \alpha_i C(x_i, x_*) \quad (1)$$

and variance

$$\hat{\sigma}^2(x_*) = C(x_*, x_*) - \mathbf{k}^T(x_*)(K + \sigma_\nu^2 I_n)^{-1} \mathbf{k}(x_*) \quad (2)$$

where  $\mathbf{k}(x_*)$  is the  $n \times 1$  vector of covariances  $(C(x_1, x_*), \dots, C(x_n, x_*))^T$  and  $\alpha = (K + \sigma_\nu^2 I_n)^{-1} \mathbf{t}$ . The prediction  $\hat{y}(x_*)$  is a linear combination of the covariance

functions  $C(x_i, x_*)$  with the coefficient  $\alpha_i$  (see (1)). The prediction variance function (2) indicates the confidence level of the model when applied to a new sample (large variance means low confidence and small variance means high confidence). If the variance is large, a predicted value should be interpreted with caution. We use the term uncertainty,  $c(x_*)$ , to denote the level of confidence for sample  $x_*$  and measured it as  $c(x_*) = 2\hat{\sigma}(x_*)$  for sample  $x_*$  throughout the paper.

Using a covariance function  $C(x, x')$ , we can directly predict the function values for novel test points. However, sometimes different problems (and data sets) are best handled with different covariance functions. To define an appropriate covariance function, we have used a parametric family of standard covariance functions, e.g. with parameters  $\Theta = [\theta_1, \dots, \theta_n]$ , followed by searching for the optimal values of  $\Theta$ . There are several ways to tune  $\Theta$ , e.g. to find  $\Theta$  which maximizes the log likelihood  $l = \log P(\mathbf{t}|\Theta) = -\frac{1}{2} \log |K + \sigma_\nu^2 I_n| - \frac{1}{2} \mathbf{t}^T (K + \sigma_\nu^2 I_n)^{-1} \mathbf{t} - \frac{n}{2} \log 2\pi$ , where  $|\cdot|$  denotes the determinant.

### C. Test reject method

One of the advantages of the Gaussian process method is that a measure of confidence can be given for individual predictions. Intuitively, a covariance function measures similarity between two samples. Larger variance of a test sample implies that it is not represented by the model based on given training samples. It is thus reasonable to either reject a predicted value with a large variance or treat it with caution, i.e. low confidence.

It is usually assumed that training and test data samples are drawn from the same distribution, i.e. that the training data is a good representative for any conceivable test data. This is not always the case for biological problems. For example, in the case of detecting potential epitopes among pathogen genomes, the training data set contains a large proportion of individual peptides with high binding affinities to MHC molecules and the test data set is a set of peptides from the whole genome of a pathogen. In such test data, there are only very few epitopes amongst a large number of possible peptides. Such possible targets may be poorly represented by training data that have been assembled to contain only confident binders and non-binders.

We do not use a fixed confidence level for all different data sets, but instead we use the prediction uncertainty distribution over the unlabelled pathogen genome to set an threshold for each model. Seeger [20] suggests that the exact uncertainty estimates are of less concern. More important is the quality of the decisions based on them. Decisions can be made on basis of expectations over predictive uncertainty distributions [20]. We present a simple threshold method based on such expectations below.

There are many more negative samples in a pathogen genome, causing the uncertainty density distribution on test samples to be skewed towards higher values. To alleviate any biases, the uncertainty threshold (at which tests are filtered out) is determined by

$$c_\theta = \frac{\min(\hat{c}_1, \dots, \hat{c}_k) + \max(\hat{c}_1, \dots, \hat{c}_k)}{2}, \quad (3)$$

where  $\hat{c}_1, \dots, \hat{c}_k$  are prediction uncertainties on the test data set  $(x_1, \dots, x_k)$ , after removing outliers from the full predictive uncertainty distribution. When the prediction uncertainty distribution is not skewed, the median value of  $\hat{c}_1, \dots, \hat{c}_k$  is used as the threshold.

Test samples with prediction uncertainties larger than the threshold are treated as if they were lacking binding affinities. Importantly, the threshold is determined from the test data set using only prediction variances, without knowledge of the targets used during training. In Gaussian process regression, uncertainty estimates do not depend on the target labels [20].

By thresholding predictions, the test reject method essentially halves the number of positive predictions. It is therefore not possible to directly use conventional classification performance metrics. However, the contribution of the prediction variance can also be evaluated by adding it to the classifiers output, enabling comparison with previous work. The impact of the variance is here weighted by a user specified coefficient,  $\delta$ . We refer to this method as a variance-weighted Gaussian process (vw-GP). Note that smaller value means positive, larger means negative.

$$\hat{y}_\delta(x_*) = \hat{y}(x_*) + \delta \hat{\sigma}^2(x_*) \quad (4)$$

### III. EXPERIMENTS

#### A. Performance comparisons on benchmark data sets

Recently, Peters *et al.* identified a benchmark MHC-peptide binding data set of variant human and mouse alleles to compare three prediction methods used in-house, including ARB, ANN, SMM and other prediction tools available online [21]. They collected 54 data sets of MHC-peptide binding affinities with different peptides length (octamers, nonamers and decamers) that bind to mouse and human MHC molecules. We focused on the 11 H-2<sup>b</sup> data sets for mouse. The data sets were downloaded from <http://mhcbindingpredictions.immuneepitope.org/dataset.html>.

For the MHC-peptide binding problem, IC<sub>50</sub> = 500nM is usually taken as a threshold to discriminate between binders and non-binders. If the wet-lab measured (IC<sub>50</sub>) value is smaller than this threshold, the corresponding peptide is considered a binder. Hence, the MHC-binding problem becomes a classification problem if all peptides are grouped accordingly.

Peters *et al.* compared their model's performance to others based on the area under the ROC (Receiver operating characteristic) curve (AUC). To assess the performance of our Gaussian process models, we first repeated the five-fold cross-validation analysis using the same training and test data splits done by Peters *et al.* For the covariance function of Gaussian processes, we used the squared exponential covariance function with isotropic distance measure [19], which is similar to the standard Gaussian/RBF kernel often used in conjunction with SVMs. To compare Gaussian processes with SVMs, the five-fold cross-validation was performed on SVMs using the Gaussian kernel as well. We used GPML (<http://www.gaussianprocess.org/gpml/code/matlab/doc/>) Matlab package by Rasmussen and Williams [19] to build Gaussian process regression models and LibSVM Matlab version 2.83 [22] to build SVM models.

The key incentive of using test rejection is to remove uncertain predictions from consideration—an aspect of classification performance not directly captured by the AUC. However, we illustrate the contribution of the variance on standard classification performance by weighting its impact on the raw classifier output for each of the benchmark data sets (4).

#### B. Epitope prediction on Vaccinia virus sequences

The benchmark data sets of Peters *et al.* are assembled to contain confident binders and non-binders. For scenarios when

the test data is limited in diversity, and perhaps similar to known binders and non-binders, the test reject method is unlikely to provide substantive assistance. To test the efficacy of a test reject method, it is thus necessary to subject the model to a much wider spectrum of candidates. As a consequence, some inputs will be less well-represented by the predictive model and will be rejected even if their prediction is firmly in the positive territory.

For a biologist, performing peptide binding experiments using a complete set of possible nonamers in a pathogen's genomic sequence is impractical. Computer models can alleviate the problem by providing a short-list of possible experimental targets. Moutaftsi *et al.* experimentally tested the value of predictive algorithms for identifying CD8+ T-cell epitopes derived from the VACV-WR strain in the H-2<sup>b</sup> mouse model [17]. With an extensive input space and limited target data, their test scenario represents a suitable challenge for a test reject method.

To build regression models, we used the same training data as used by Moutaftsi *et al.* [17]. It contains 455 octamers (8-mers) and 171 nonamers (9-mers) for H-2 K<sup>b</sup> allele, 244 nonamers and 90 decamers (10-mers) for H-2 D<sup>b</sup> allele. For the blind test data set, we used all possible 58484 octamers (K<sup>b</sup>) molecules, 58226 nonamers (D<sup>b</sup> and K<sup>b</sup>), and 57968 decamers (D<sup>b</sup>) from the full VACV-WR sequence. We compared the Gaussian process regression with and without the proposed test reject method.

Based on a consensus model involving an SMM, Moutaftsi *et al.* identified epitopes accounting for the majority of the CD8+ T-cell responses evoked from VACV-WR infection. They found 18 octamer and 9 nonamer epitopes for K<sup>b</sup> allele and 18 nonamer and 4 decamer epitopes for D<sup>b</sup> allele.

We note that there are more peptides with lower binding affinities (higher IC<sub>50</sub> values) in the four training data sets. Specifically, for the D<sup>b</sup> allele and decamer group there are only nine decamers with IC<sub>50</sub> values less than log<sub>10</sub>(500). Due to experimental detection limitations, the IC<sub>50</sub> value can not be determined if it is beyond a given threshold. In all the training data sets, constant values were used to threshold such peptides. For the K<sup>b</sup> octamer group, there are 107 binders and 348 non-binders; for the K<sup>b</sup> nonamer group, there are 38 binders and 133 non-binders; for the D<sup>b</sup> nonamer group, there are 43 binders and 201 non-binders; and for the D<sup>b</sup> decamer group, there are 9 binders and 81 non-binders.

A binding affinity predictor was developed on each of the four data sets using Gaussian process regression with the squared exponential covariance function with isotropic distance measure. All simulations were performed within groups (H-2 allele and peptide length combination). The four training data sets do not overlap with the VACV-WR sequence. Instead of using a consensus method as Moutaftsi *et al.*, we employed the test reject method to remove peptides in the VACV-WR sequence with greater variance values on predicted binding affinities than the rejection threshold (as per Equation 3) and then ranked all remaining peptides.

### IV. RESULTS

#### A. Cross-validation performance of Gaussian processes and SVM

To establish the overall MHC class I-peptide binding prediction accuracy of Gaussian processes, we performed cross-validation simulations on 11 public benchmark data sets [21]. The classification accuracy comparison is shown in Table I, where AUC values for ARB, SMM and ANN were taken from [21]. We

performed the SVM and Gaussian process simulations using identical data set splits. Since ANN models were tested on the data sets containing only nonamers, only results of ANN for nonamers are listed.

TABLE I

CROSS-VALIDATION RESULTS (AUC VALUES) ON ELEVEN DIFFERENT MOUSE ALLELE DATA SETS USING ARB, SMM, ANN, SVM, GP AND A VARIANCE-WEIGHTED GP.

Allele + length	Size	ARB	SMM	ANN	SVM	GP	vw-GP
H-2 D <sup>b</sup> 9	303	0.865	0.912	0.933	0.932	0.925	0.926
H-2 D <sup>b</sup> 10	134	0.715	0.759		0.708	0.750	0.765
H-2 D <sup>d</sup> 9	85	0.696	0.853	0.925	0.887	0.916	0.929
H-2 D <sup>d</sup> 10	75	0.990	0.997		1.000	0.999	0.998
H-2 K <sup>b</sup> 8	480	0.846	0.890		0.904	0.895	0.898
H-2 K <sup>b</sup> 9	223	0.792	0.810	0.850	0.740	0.796	0.798
H-2 K <sup>d</sup> 9	176	0.798	0.936	0.939	0.906	0.929	0.932
H-2 K <sup>d</sup> 10	70	0.486	0.576		0.457	0.493	0.520
H-2 K <sup>k</sup> 8	80	0.782	0.893		0.915	0.910	0.914
H-2 K <sup>k</sup> 9	164	0.758	0.770	0.790	0.793	0.808	0.810
H-2 K <sup>k</sup> 10	57	0.615	0.576		0.741	0.643	0.656

From the cross-validation results (AUC values), the ARB method is inferior to other methods and the average accuracies of SMM, Gaussian processes and SVMs on the 11 data sets are very similar. Peters *et al.* only used five-fold cross-validation once, so standard deviation on performance is not available for ARB, ANN and SMM methods. However, we repeated the five-fold cross-validation ten times with the Gaussian processes on K<sup>b</sup> 8-mers, K<sup>b</sup> 9-mers, D<sup>b</sup> 9-mers and D<sup>b</sup> 10-mers to check performance deviations. We got the mean AUC value and its standard deviation for K<sup>b</sup> 8-mers as 0.891 (0.006); 0.792 (0.012) for K<sup>b</sup> 9-mers; 0.915 (0.005) for D<sup>b</sup> 9-mers and 0.757 (0.048) for D<sup>b</sup> 10-mers. The performance of Gaussian process regression is stable between different data splits.

In some cases, the ANN performance is slightly better. However, several parameters need to be tuned separately (e.g. number of hidden nodes and stopping criterion) increasing the risk of a parameter selection bias. In contrast, when the covariance function has been chosen, the Gaussian process requires no further tuning.

Overall, Gaussian processes are not inferior to any of the other methods on the standard MHC-peptide binding problems. We thus expect to get the same classification accuracy with Gaussian processes as with ANN or SVM on any similar data sets.

To illustrate the impact of uncertainty in a standard classification context, a variance-weighted Gaussian process was trained and tested under the same conditions as the other methods (last column in Table I). We note that the AUC increases with larger  $\delta$  on the mouse benchmark data sets up to about  $\delta = 20$  (data shown in the table).

### B. Using test reject method to detect a subset of T-cell epitopes

We used MHC-peptide binding affinity predictors built with Gaussian process regression methods on the four training data sets to predict all the possible peptides for each combination of H-2 allele and peptide length. Figure 1 displays the estimated density function of prediction uncertainty values for each of the four data sets (after removing outliers). We can see that in each case, the

function is skewed towards higher uncertainties. The predictor-specific threshold is determined from these by using equation (3) to separate trusted from rejected predictions.

Figure 2 displays the predicted  $\log_{10}$  IC<sub>50</sub> values and their corresponding prediction uncertainties for each of the four data sets. The experimentally verified epitopes (the targets we are interested in identifying) are marked as crosses. The coloring in  $y$ -direction separates them into trusted and rejected predictions based on the predictor-specific threshold. It is interesting to note that non-binders are more often predicted with higher uncertainties, in spite of there being more non-binder than binder samples in the training data sets. This observation could be explained by experimental detection limitations for non-binders. The prediction uncertainties may thus reflect sample measurement noise.

Figure 3 compares the ability of finding experimentally verified epitopes with and without the proposed test reject method for each of the four data sets. The curves with dashed lines illustrate the true positive rate when using Gaussian process regression method only. The curves with solid lines represent the true positive rate using the Gaussian process with test rejection. We refer to these curves as epitope match curves. The  $x$ -axis of the epitope match curves is the number of top-ranked peptides (sorted by predicted  $\log_{10}$  IC<sub>50</sub> values in ascending order). The first peptide is thus most likely to be an epitope. The  $y$ -axis of the epitope match curves represents the number of true epitopes amongst the top-ranked peptides (true epitopes as designated as such in the data sets of Moutaftsi *et al.*).

From the four epitope match curves, it is clear that using the test reject method removes several false positives without sacrificing the sensitivity of the original method. Taking the K<sup>b</sup> allele and octamer group as an example, by using the test reject method, the sensitivity increases sharply in the beginning and there are seven true epitope matches out of 55 top predicted binders. After about 100 top predicted binders, the epitope match curve goes up quickly and illustrates 17 hits out of 255 predicted binders, and finally 18 hits out of predicted 452 binders.

## V. DISCUSSION AND CONCLUSION

It is interesting to compare the performance of Gaussian process regression with the test reject method to the consensus method used by Moutaftsi *et al.* on the VACV-WR sequence. We first briefly discuss the test reject method as applied on the complete genomic sequence and then on a sub set that allows a direct and thorough evaluation against the consensus method.

The Gaussian process with the test reject method performs well on the four data sets but is out-matched by the consensus method on the D<sup>b</sup> alleles. For the K<sup>b</sup> allele octamers, the test reject method generally improves on the Moutaftsi *et al.* consensus method, especially after the 100 top-ranked epitopes. For K<sup>b</sup> nonamers and D<sup>b</sup> decamers, the rate of true epitopes is initially higher for the test reject method. Here, the consensus method shows better performance when more epitopes are included.

Due to the special CD8+ T-cell epitope experiment design the performance of the Gaussian processes with test reject method is likely underestimated. Moutaftsi *et al.* first collected all possible octamers, nonamers and decamers from the VACV-WR sequence. They then applied four scoring-matrix prediction methods (Udaka, Parker, ARB and SMM) on the all peptides, except for decamers for which only ARB and SMM models were available. They ranked peptides according to their prediction scores. For each

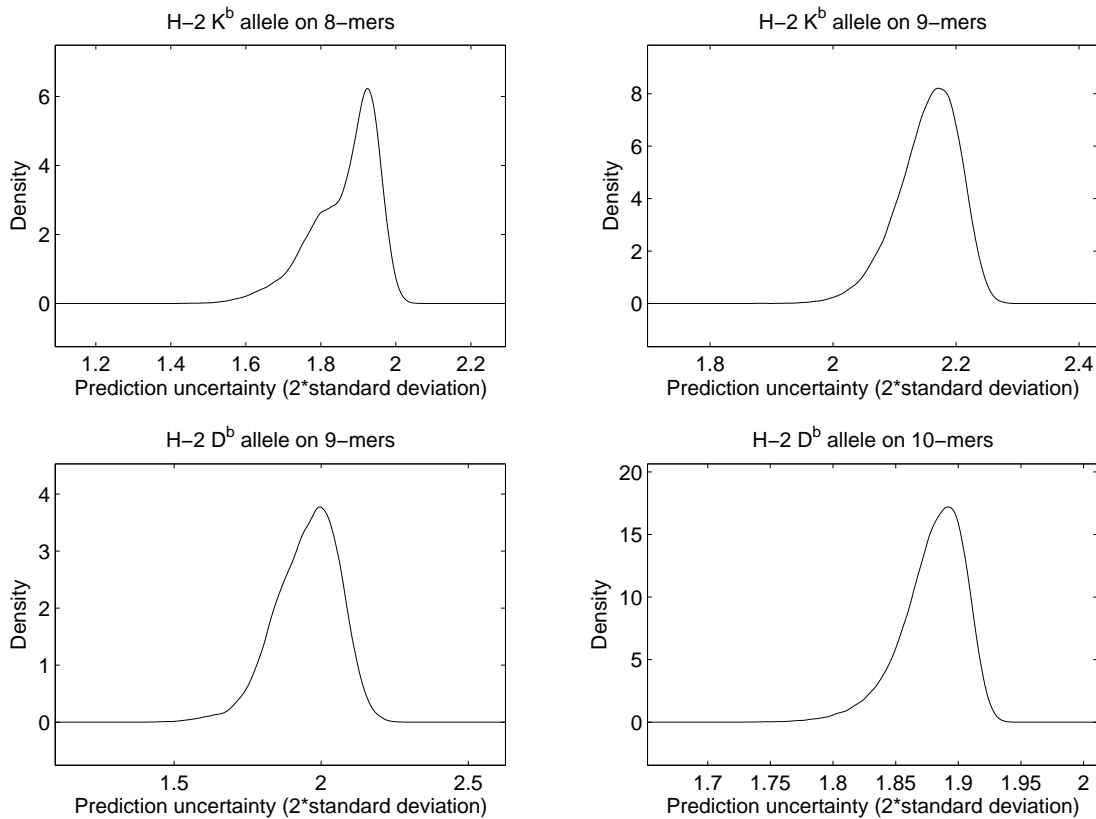


Fig. 1. Prediction uncertainty densities for the four models on the complete VACV-WR genome.

allele and peptide length combination, they assigned each peptide the median of the rank values determined in the individual binding predictions. Based on the consensus prediction, they selected the peptides in the top 1% median ranks for their study, comprising 564 peptide for each group, not from all possible peptides (about 58000). Finally, they measured antigenicity only for the 564 peptides in each group. This way, the experiments were biased towards predictable epitopes, leaving the possibility of additional true epitopes.

In the Moutaftsi *et al.* study, the test data set in each group contained only 564 peptides. We scanned the complete genomic sequence, all possible peptides, which is 100 times larger than the test data sets they used. In other words, our top-ranking predictions are likely to contain additional true epitopes not yet validated experimentally.

To resolve the effect arising from data selection bias, we reused our models on the smaller test data sets containing 564 peptides in each group (henceforth the “sub-564 data set”) and compared the performance with or without test reject method to Moutaftsi *et al.*’s consensus method.

Figure 4 shows the prediction uncertainty densities of the four MHC molecule and peptide length groups on the sub-564 data set. Compared to Figure 1, the uncertainty distribution on this smaller data set is less skewed. The median value of uncertainties was thus used as the threshold for the group of  $K^b$  8-mers,  $K^b$  9-mers and  $D^b$  9-mers. For the group  $D^b$  10-mers, Equation 3 was used to set the predictor-specific threshold.

The epitope match curves for each of the three methods, GP

with test reject method (TRM), the GP without TRM and the Consensus method, when applied on the sub-564 data set are presented in Figure 5.

We performed a bootstrap test on each of the three methods by re-sampling 75% (without replacement [23]) of the 564 peptides for each allele and peptide length group. Each test resulted in new sub-set of inspected peptides, for each required number of true epitopes. The test was repeated 100 times with different sub-sets. Tables II, III, IV and V provide the average number of peptides that need to be inspected before the specified count of true epitopes is reached.

The limited size of the sub-564 data set begs further corroboration of the difference. To provide a notion of statistical significance, we present two tests. First, we performed a simple binomial test on all the 47 true epitope counts for which any of the three methods has at least one run with a result. That is, we investigate the null hypothesis that a pair of methods is randomly superior in each column of the tables. The one-tailed  $p$ -value for GP with TRM vs GP without TRM (using the tail favoring GP with TRM) is less than  $10^{-7}$ . The corresponding value for TRM vs Consensus is less than  $10^{-2}$ . Both comparisons thus show that the TRM method significantly outperforms each of the other two methods. We note that the binomial test is compromised by the columns not being independent.

Resampling the data provides us with three means of number of predictions, for each true epitope count. A standard paired  $t$ -test was used to compare the means produced by each of the methods, for each true epitope count. By hypothesizing that the means are

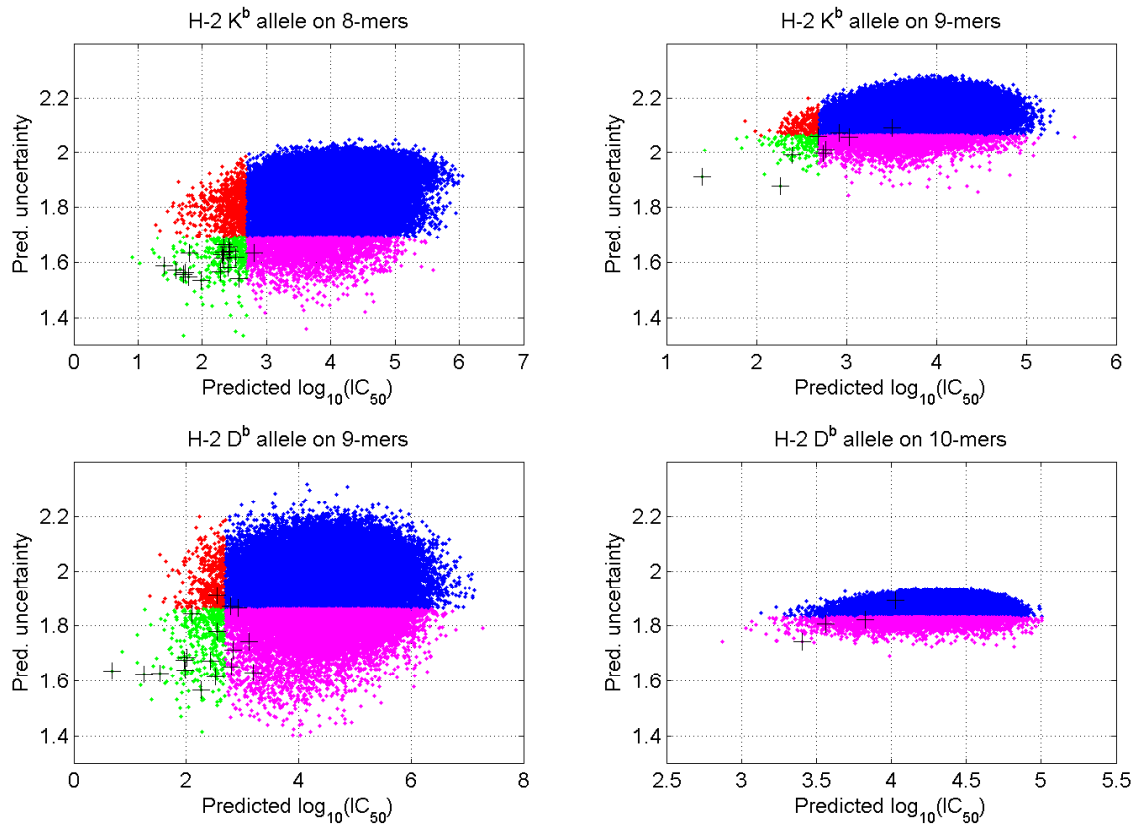


Fig. 2. Prediction uncertainties and predicted affinities scatter plot using the four models on the complete VACV-WR genome. Predictions for known epitopes are marked as crosses. Trusted and rejected predictions, based on the predictor-specific threshold, are delineated by the coloring in  $y$ -direction. The standard IC<sub>50</sub> threshold is indicated by the coloring in  $x$ -direction.

the same for each pair of methods, we determine the one-tailed  $p$ -value (again the tail favoring the TRM method to illustrate the superior specificity of this method). For each true epitope count in the tables, we asterisk if the test shows a  $p$ -value less than 0.05 (significance at the 5% level). On a cautionary note, the independence assumption of this test is compromised since the sample sets overlap. However, the collective evidence in favour of the test reject method is convincing, at least when more than a couple of true epitopes are required to be found.

The analyses on the smaller data sets confirm that test rejection generally improves the ratio of true positives in the predicted set. Comparing to the consensus method, with test rejection Gaussian process regression method generally finds true epitopes earlier, effectively reducing the need for expensive and time-consuming wet-lab experiments.

An additional issue is the transparency of the consensus method. Three of the four methods (Udaka, Parker and ARB) used different and to some extent additional training data which impact we can not readily assess. Generally, when using consensus methods, the designer needs to determine how and which individual methods to be included when processing novel inputs. In addition, the output of the individual methods might have different meanings, an issue that Moutaftsi *et al.* circumvented by using individual ranking. In contrast, we employed an approach based on Gaussian processes which offer a theoretical framework for decision making and exploits an inherent feature of such

methods to evaluate each prediction confidence.

Finally, we return to the benchmark data sets of Peters *et al.*, including in total 89 different allele/length combinations for human, mouse, macaque and chimpanzee. For each combination we build a model using Gaussian processes using the same data set splits as used by Peters *et al.* The numbers of samples in the human benchmark *training* sets are generally much greater than in any of the other species (mean 779 and 154 for human and mouse, respectively), leading to substantial differences in distribution of prediction variance on benchmark *test* data.

Figure 6 shows the estimated densities of prediction variance collected for three groups of test data: human and mouse alleles (from the Peters *et al.* benchmark sets) and the complete VACV-WR nonamer set. (Macaque and chimpanzee variances occupy the space between human and mouse.) A couple of things are worth noting. First, the human benchmark sets render much smaller prediction variances. This indicates that the human allele classifiers make confident predictions. Importantly, it implies that the setting of the TRM threshold,  $c_\theta$  (3), is low. Stringent  $c_\theta$  settings derived from training on human data would, if applied in the mouse TRM, reject almost all predictions on mouse test data. Conversely, a mouse-derived cut-off would reject very few predictions on human data.

Second, the density of VACV-WR is based on about 58000 peptides, most of which are negatives and dissimilar to the training data. Indeed, the density estimate in Figure 6 occupies a

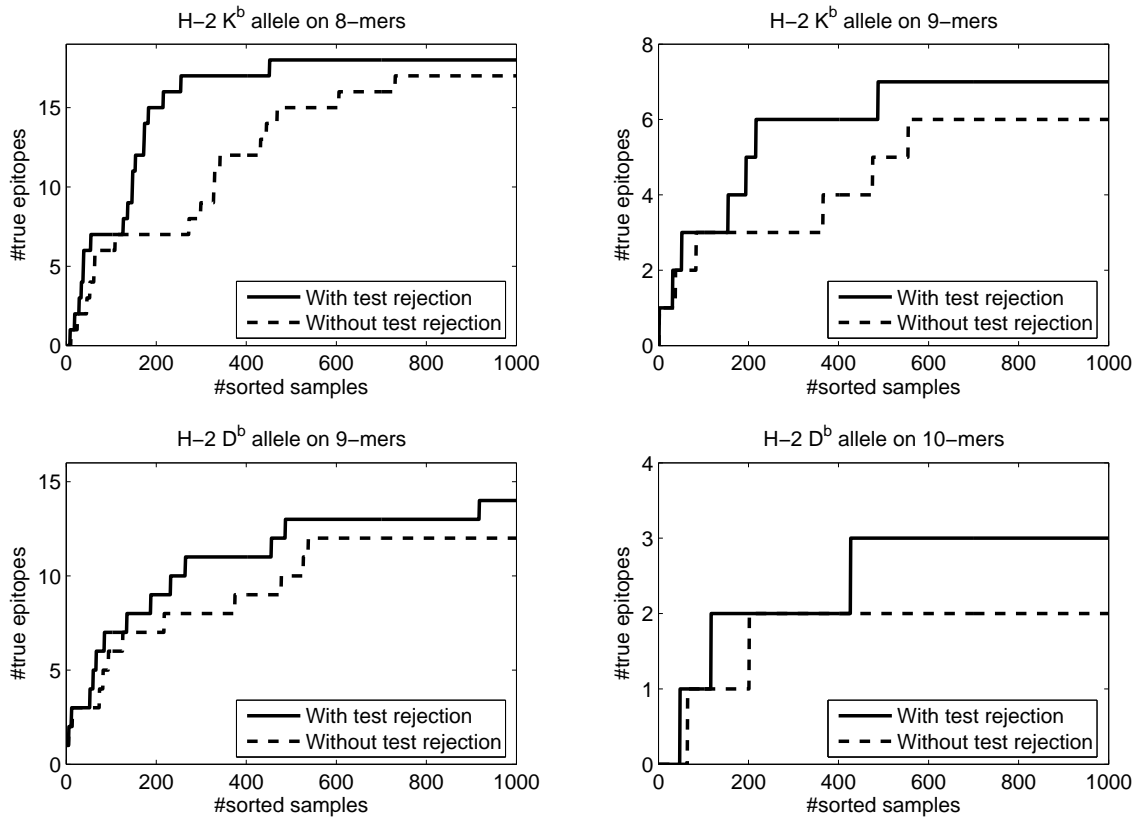


Fig. 3. Epitope match curves using Gaussian process regression with and without test reject method for the four models. The match curve displays the number of peptides needed to be predicted ( $x$ -axis) in order to find the corresponding number of epitopes ( $y$ -axis).

TABLE II  
EPILOPE MATCH LISTS OF H-2 K<sup>b</sup> AND 8-MERS GROUP IN THE SUB-564 DATA SET.

Epitopes found	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
w TRM	8	16	22	28	43	59	76	86	92	98	108	116	126	138	150	149	180	-	
w/o TRM	10	20	29	37	61	89	118	136	146	161	181	198	225	248	279	261	339	-	
Consensus	4	14	24	31	40	50	67	95	125	154	174	186	194	197	204	206	210	-	
w vs w/o TRM	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	-
TRM vs Consensus			*	*				*	*	*	*	*	*	*	*	*	*	*	-

Gaussian process with and without TRM (“w TRM” and “w/o TRM”, respectively), and the Consensus method of Moutafsi *et al.* are compared. Cells (in rows 1-3) contain the number of predictions required to find the specified number of epitopes. Cells in rows 4-5 are asterisked if GP with TRM is better than the alternative method ( $p < 0.05$ ) at that epitope-count. ‘-’ marks that the value can not be determined.

TABLE III  
EPILOPE MATCH LISTS OF H-2 K<sup>b</sup> AND 9-MERS GROUP IN THE SUB-564 DATA SET

Epitopes found	1	2	3	4	5	6	7	8	9
w TRM	11	40	79	116	138	154	169	174	-
w/o TRM	13	53	122	194	248	304	347	384	417
Consensus	80	191	254	299	306	311	317	321	325
w vs w/o TRM	*	*	*	*	*	*	*	*	*
TRM vs Consensus	*	*	*	*	*	*	*	*	*

Cells (in rows 1-3) contain the number of predictions required to find the specified number of epitopes. Cells in rows 4-5 are asterisked if GP with TRM is better than the alternative method ( $p < 0.05$ ) at that epitope-count. ‘-’ marks that the value can not be determined.

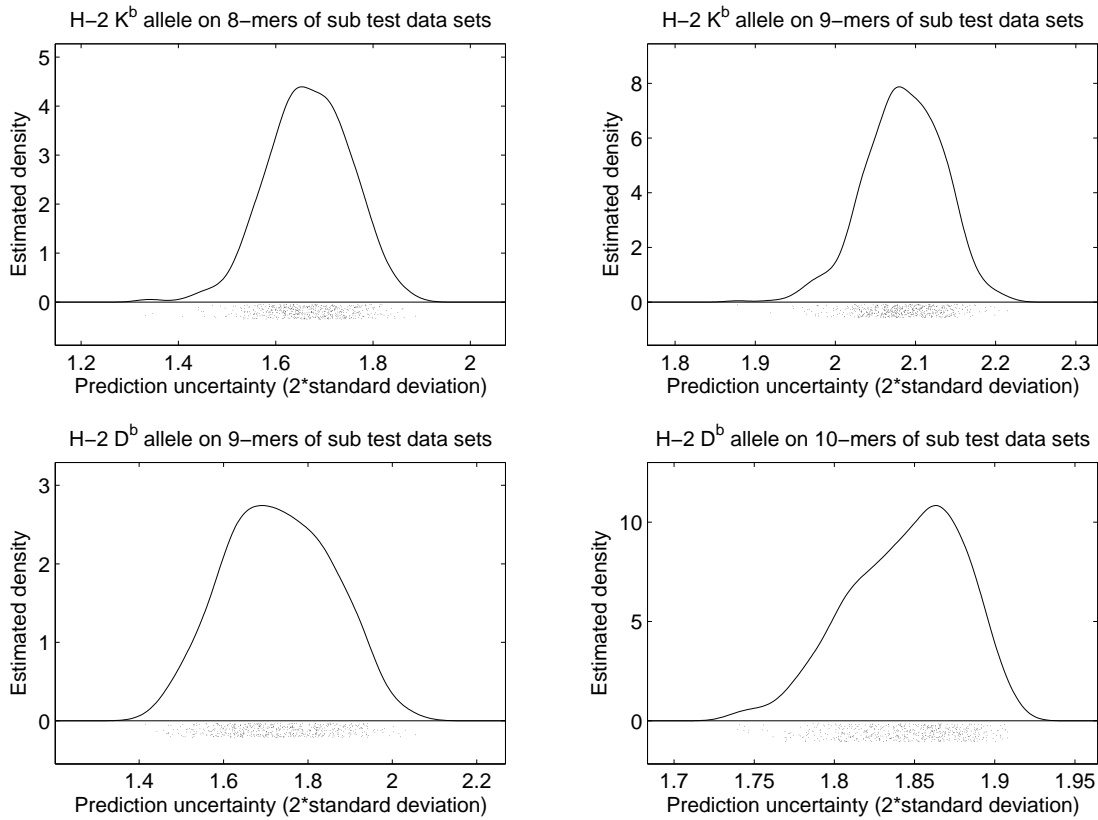


Fig. 4. Prediction uncertainty densities of the four models on the sub-564 data set.

TABLE IV  
 EPIPE MATCH LISTS OF H-2 D<sup>b</sup> AND 9-MERS GROUP IN THE SUB-564 DATA SET

Epitopes found	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
w TRM	2	8	18	29	41	59	82	103	116	132	146	155	-	-	-	-	-	-
w/o TRM	3	11	27	45	61	89	119	149	179	205	228	260	290	304	321	329	335	-
Consensus	1	4	13	30	47	68	91	117	146	170	209	268	312	337	355	375	378	-
w vs w/o TRM	*	*	*	*	*	*	*	*	*	*	*		-	-	-	-	-	-
TRM vs Consensus					*	*	*	*	*	*	*		-	-	-	-	-	-

Cells (in rows 1-3) contain the number of predictions required to find the specified number of epitopes. Cells in rows 4-5 are asterisked if GP with TRM is better than the alternative method ( $p < 0.05$ ) at that epitope-count. '-' marks that the value can not be determined.

TABLE V  
 EPIPE MATCH LISTS OF H-2 D<sup>b</sup> AND 10-MERS GROUP IN THE SUB-564 DATA SET

Epitopes found	1	2	3	4
w TRM	31	74	111	-
w/o TRM	47	176	340	406
Consensus	78	123	163	178
w vs w/o TRM	*	*	*	
TRM vs Consensus	*	*	*	

Cells (in rows 1-3) contain the number of predictions required to find the specified number of epitopes. Cells in rows 4-5 are asterisked if GP with TRM is better than the alternative method ( $p < 0.05$ ) at that epitope-count. '-' marks that the value can not be determined.

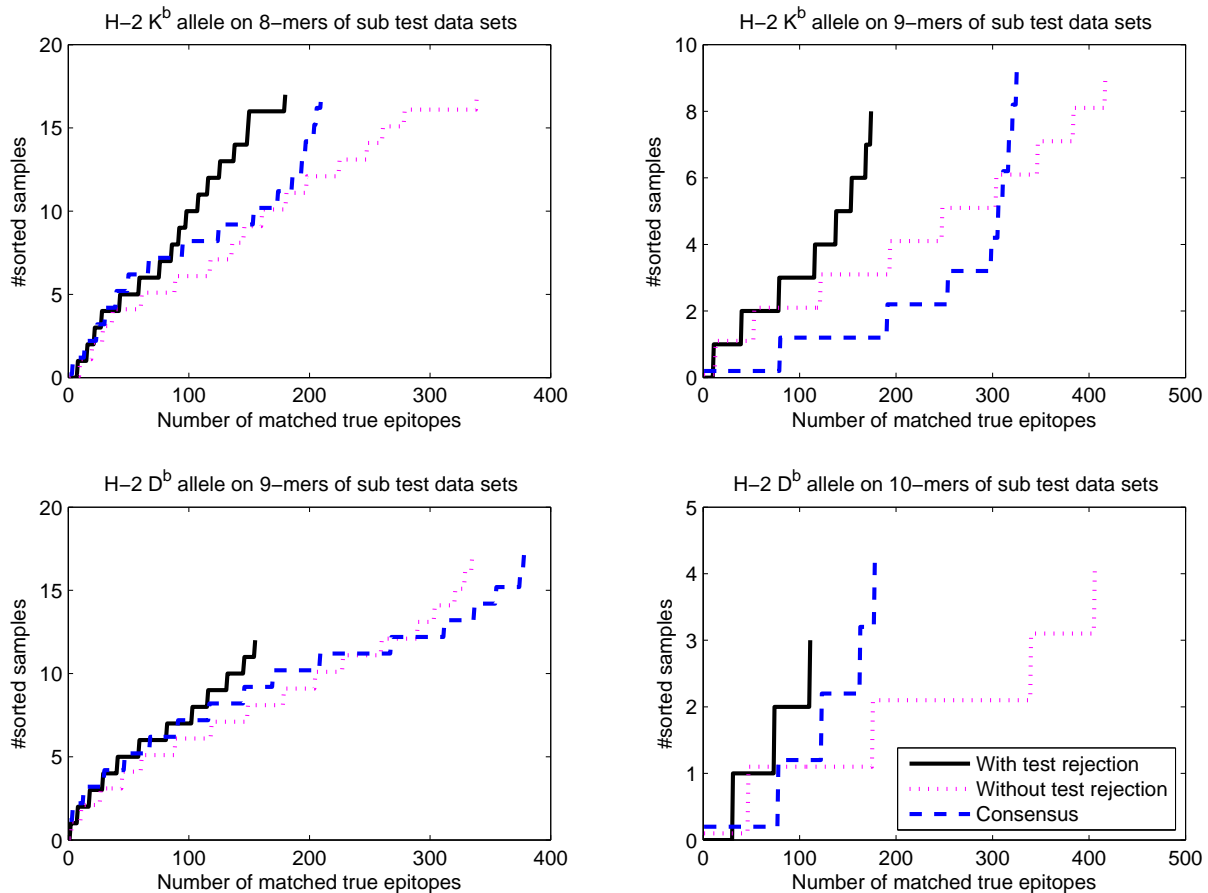


Fig. 5. Epitope match curves using Gaussian processes with or without test reject method and a consensus method on the sub-564 test data set. The match curve displays the *number* of peptides needed to be predicted (*x*-axis) in order to find the corresponding *number* of epitopes (*y*-axis).

region upwards relative both human and mouse benchmark sets. Removing all but the high-scoring candidates as identified in the Moutaftsi *et al.* study (i.e. the sub-564 set), the density has a mean of 0.039 (occupying a middle-ground between human 0.018 and the full VACV-WR sequence 0.052; data not shown).

With the aforementioned analysis in mind, the benchmark data sets still supply us with an additional resource to evaluate the ability of the test reject method to flag false positives. For every allele and length we use the corresponding GP model to monitor the strongest predictions and compare with the test rejection method. As before, TRM removes predictions per (3). Specifically, the number of false positives before the first true positive is determined. Grouped by host organism, mouse alleles render eight and seven false positives, for standard GP and TRM, respectively. For human alleles, a total of 26 false positives by standard GP and 13 by TRM. For macaque, the numbers are 109 and 45, respectively. No false positives are observed before the first true positive on chimpanzee alleles, for either method.

The ability of TRM to remove false positives and not true positives drops further down the ranked list. Specifically, we looked at the five strongest predictions for each allele and counted the number of true positives among them. Both standard GP and TRM render 49 true positives for mouse alleles (of a total 65). GP hits 229 and TRM 224 for human alleles (max 265), 74 vs. 73

for macaque (max 100), and 10 vs. 10 for chimpanzee (max 10). At least for human alleles, the TRM threshold ( $c_\theta$ ) appears to remove too many positive predictions from consideration and may need to be relaxed to the level of the more successful mouse settings. The current setting of  $c_\theta$  considers only the prediction variance in relative terms (cf. 3). Future work should consider a setting informed by the absolute range of variance or by gleaning a complete pathogen sequence.

We also tried the variance-weighted Gaussian process on the human, macaque and chimpanzee alleles [21]. There was no increase in AUC for any value of  $\delta$  on human and macaque. There was a minor increase in mean AUC (from 0.870 to 0.880) for chimpanzee alleles when  $\delta \approx 50$ . In summary, the observations on the benchmark data sets indicate that vw-GP and TRM usually outperform standard GP when the absolute prediction variance of test data is high. For data with consistently low prediction variance, the standard method appears to work as well—or better.

In this study, we verified that Gaussian process regression method can be used to predict MHC class I-peptide binding with similar accuracy as the best alternative methods. Furthermore, we proposed and evaluated a test reject method to increase the true positive ratio in a set of predictions. The use of confidence measures in *in silico* prediction models is promising for large-scale and realistically diverse sequence problems where the number of

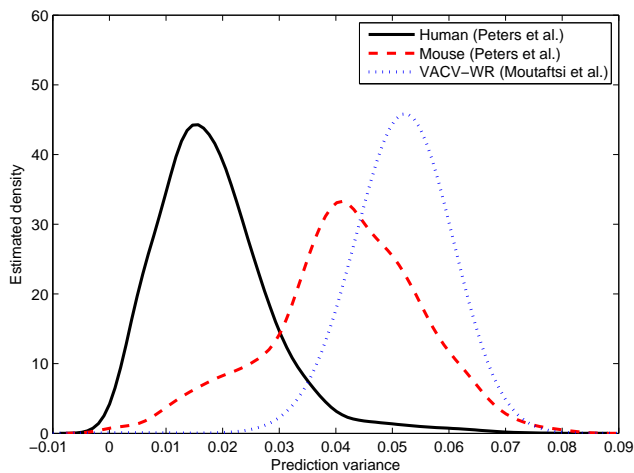


Fig. 6. Probability density estimates of the absolute prediction variance for three groups of test data: all human and mouse alleles in the Peters *et al.* benchmark sets, and the complete VACV-WR nonamer set taken from the Moutaftsi *et al.* study. The means are  $0.018 \pm 0.010$ ,  $0.042 \pm 0.014$  and  $0.052 \pm 0.006$ , for human, mouse and VACV-WR respectively. Each density was estimated from test data by using the GP tuned from the corresponding allele/length training data then grouping them according to organism. For VACV-WR we used only mouse D<sup>b</sup> 9 training data. Smoothing was performed using a standard Gaussian kernel (width=0.005).

false positives is large. Indeed, for complete pathogen sequences, this is usually the case. However, our tests on human alleles—as represented in assembled benchmark sets—indicate that the setting of the threshold requires careful attention.

#### ACKNOWLEDGMENT

LY was funded within the National Research School in Genomics and Bioinformatics hosted by Göteborg University, Sweden. This project was supported in part (LY and VB) by the ImmunoGrid project, under EC contract FP6-2004-IST-4, No. 028069. MB was supported by the ARC Centre of Excellence in Bioinformatics. The authors are grateful for constructive advice from three anonymous reviewers and Dr. Timothy Bailey.

#### REFERENCES

- [1] J. W. Yewdell and J. R. Bennink, "Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses," *Annu. Rev. Immunol.*, vol. 17, pp. 51–88, 1999.
- [2] N. Zaitlen, M. Reyes-Gomez, D. Heckerman, and N. Jovic, "Shift-invariant adaptive double threading: Learning MHC II - peptide binding," *Lecture Notes in Computer Science*, vol. 4453, 2007.
- [3] P. A. Reche, J.-P. Gluttinga, and E. L. Reinherz, "Prediction of MHC Class I binding peptides using profile motifs," *Human Immunology*, vol. 63, pp. 701–709, 2002.
- [4] H. H. Bui, J. Sidney, B. Peters, M. Sathiamurthy, A. Sinichi, K. A. Purton, B. R. Mothe, F. V. Chisari, D. I. Watkins, and A. Sette, "Automated generation and evaluation of specific MHC binding predictive tools: ARB matrix applications," *Immunogenetics*, vol. 57, pp. 304–14, 2005.
- [5] B. Peters, W. Tong, J. Sidney, A. Sette, and Z. Weng, "Examining the independent binding assumption for binding of peptide epitopes to MHC-I molecules," *Bioinformatics*, vol. 19, pp. 1765–1772, 2003.
- [6] B. Peters and A. Sette, "Generating quantitative models describing the sequence specificity of biological processes with the stabilized matrix method," *BMC Bioinformatics*, vol. 6, 2005.

- [7] K. Udaka, K. H. Wiesmuller, S. Kienle, G. Jung, H. Tamamura, H. Yamagishi, K. Okumura, P. Walden, T. Suto, and T. Kawasaki, "An automated prediction of MHC class I-binding peptides based on positional scanning with peptide libraries," *Immunogenetics*, vol. 51, pp. 816–828, 2000.
- [8] K. C. Parker, M. A. Bednarek, and J. E. Coligan, "Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains," *J Immunol*, vol. 152, pp. 163–75, 1994.
- [9] M. C. Honeyman, V. Brusnic, N. L. Stone, and L. C. Harrison, "Neural network-based prediction of candidate T-cell epitopes," *Nat Biotechnol*, vol. 16, no. 10, pp. 966–969, Oct 1998.
- [10] M. Nielsen, C. Lundegaard, P. Worning, S. L. Lauemoller, K. Lamberth, S. Buus, S. Brunak, and O. Lund, "Reliable prediction of T-cell epitopes using neural networks with novel sequence representations," *Protein Sci.*, vol. 12, pp. 1007–1017, 2003.
- [11] P. Dönnes and A. Elofsson, "Prediction of MHC class I binding peptides, using SVMHC," *BMC Bioinformatics*, vol. 3, 2002.
- [12] Y. Zhao, C. Pinilla, D. Valmori, R. Martin, and R. Simon, "Application of support vector machines for T-cell epitopes prediction," *Bioinformatics*, vol. 19, pp. 1978–1984, 2003.
- [13] H. Riedesel, B. Kolbeck, O. Schmetzer, and E. W. Knapp, "Peptide binding at class I major histocompatibility complex scored with linear functions and support vector machines," *Genome Informatics*, vol. 15, pp. 198–212, 2004.
- [14] G. L. Zhang, I. Bozic, C. K. Kwok, J. T. August, and V. Brusnic, "Prediction of supertypespecific HLA class I binding peptides using support vector machines," *Journal of Immunological Methods*, vol. 320, pp. 143–154, 2007.
- [15] H. Mamitsuka, "Predicting peptides that bind to MHC molecules using supervised learning of hidden Markov models," *Proteins*, vol. 33, pp. 460–474, 1998.
- [16] T. Hertz and C. Yanover, "PepDist: a new framework for protein-peptide binding prediction based on learning peptide distance functions," *BMC Bioinformatics*, vol. 7(Suppl 1), p. S3, 2006.
- [17] M. Moutaftsi, B. Peters, V. Pasquetto, D. C. Tschärke, J. Sidney, H. H. Bui, H. Grey, and A. Sette, "A consensus epitope prediction approach identifies the breadth of murine T(CD8+)-cell responses to vaccinia virus," *Nat Biotechnol*, vol. 24, pp. 817–819, 2006.
- [18] D. J. C. MacKay, "Introduction to Gaussian processes," in *Neural networks and machine learning*, ser. NATO ASI, C. M. Bishop, Ed. Springer, Berlin: Springer-Verlag, 1998, vol. 168, pp. 133–165.
- [19] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, 1st ed. Cambridge, MA: The MIT Press, 2006.
- [20] M. Seeger, "Gaussian processes for machine learning," *International Journal of Neural Systems*, vol. 14, pp. 69–106, 2004.
- [21] B. Peters, H. H. Bui, S. Frankild, M. Nielson, C. Lundegaard, E. Kostem, D. Basch, K. Lamberth, M. Harndahl, W. Fleri, S. S. Wilson, J. Sidney, O. Lund, S. Buus, and A. Sette, "A community resource benchmarking predictions of peptide binding to MHC-I molecules," *PLoS Comput Biol*, vol. 2, pp. 574–584, 2006.
- [22] C. C. Chang and C. J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [23] P. Rao and M. J. Katzoff, "Bootstrap for finite populations," *Communications in Statistics - Simulation and Computation*, vol. 25, pp. 979–994, 1996.