JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 14, Number 6, 2007

Pp. 736–746
DOI: 10.1089/cmb.2007.R013

Leveraging Information Across HLA Alleles/Supertypes Improves Epitope Prediction

DAVID HECKERMAN, CARL KADIE, and JENNIFER LISTGARTEN

ABSTRACT

We present a model for predicting HLA class I restricted CTL epitopes. In contrast to almost all other work in this area, we train a single model on epitopes from all HLA alleles and supertypes, yet retain the ability to make epitope predictions for specific HLA alleles. We are therefore able to leverage data across all HLA alleles and/or their supertypes, automatically learning what information should be shared and also how to combine allele-specific, supertype-specific, and global information in a principled way. We show that this leveraging can improve prediction of epitopes having HLA alleles with known supertypes, and dramatically increases our ability to predict epitopes having alleles which do not fall into any of the known supertypes. Our model, which is based on logistic regression, is simple to implement and understand, is solved by finding a single global maximum, and is more accurate (to our knowledge) than any other model.

Key words: classifier, epitope, features, multi-task, prediction.

1. INTRODUCTION

THE HUMAN ADAPTIVE IMMUNE RESPONSE is composed of two core elements: antibody-mediated response (sometimes called humoral response) and T-cell-mediated response (sometimes called cellular response). To date, essentially all successful human vaccines have been made by exploiting the underlying mechanisms of the antibody-mediated response, for example with diseases such as polio and measles. However, for these diseases, it was known that people could recover upon acquisition of humoral immunity. In contrast, for certain viruses—for example, HIV—there are no known documented cases of a person recovering from the infection, and it is highly unlikely that the same principles of vaccine design could be successfully applied in these cases. In particular, it is thought that vaccines for diseases such as HIV must prime the cellular immune response rather than or in addition to the humoral response in order to be successful (Parham, 2004; McMichael and Hanke, 2002).

At the core of cellular response is the ability of certain antigen-presenting cells to ingest and digest viral proteins into smaller peptides, and then to present these peptides, known as *epitopes*, at the surface of the cell. This process is mediated by HLA (Human Leukocyte Antigen) molecules which form a complex with the epitope before it is presented. The epitope/HLA complexes can then be recognized by a T-cell, thereby activating the T-cell to subsequently recognize and kill virally infected cells. Several types of T-cells exist,

Microsoft Research, Redmond, Washington.

each playing its own role. In ongoing HIV vaccine research, the elicitation of a $CD8^+$ T-cell response has shown promise. Since $CD8^+$ T-cells recognize only HLA class I bound epitopes (which range in length from eight to eleven amino acids), our application focuses on such epitopes. Furthermore, we concentrate on the prediction of 9mer epitopes, as this length is the most common.

Due to specificity in a number of sequential mechanisms, only certain epitopes are both presented at the surface of antigen-presenting cells and then subsequently recognized by T-cells. This specificity is determined in part by the sequence and properties of the presented epitope and by the genetic background (i.e., allelelic diversity) of the host (humans have up to six HLA class I alleles arising from the A, B, and C loci). A crucial task in vaccine development is the identification of epitopes and the alleles that present them, since it is thought that a good vaccine will include a robust set of epitopes (robust in the sense of broad coverage and of covering regions that are essential for viral fitness in a given population characterized by a particular distribution of HLA alleles). Because experiments required to prove that a peptide is an epitope for a particular HLA allele (Goulder et al., 2001) are time-consuming and expensive, epitope prediction can be of tremendous help in identifying new potential epitopes whose identity can then be confirmed experimentally. Beyond vaccine design, epitope prediction may have important applications such as predicting infectious disease susceptibility and transplantation success.

In this work, we present a logistic regression (LR) model for epitope prediction which is more accurate than the most accurate model that we can find in the literature—DistBoost (Yanover and Hertz, 2005), and also has several practical advantages: (1) it is a well known model with many readily-available implementations, (2) its output is easy to interpret, (3) training requires $O(N)$ memory whereas DistBoost requires $O(N^2)$ memory, where N is the sample size of the data, (4) the parameters of LR given data have a single, globally optimal value that is easily learned (in contrast to DistBoost and artificial-neural-network based predictors such as NetMHC (Buus et al., 2003), which have many hidden units), and (5) it produces probabilities that tend to be well calibrated (Platt, 1999) and hence useful for making decisions about, for example, whether to confirm a prediction in the lab.

Another important contribution of this paper is that we show how to leverage information across multiple HLA alleles to improve predictive accuracy for a specific allele. An epitope is defined with respect to one or more HLA alleles. That is, a peptide which is an epitope for HLA-allele X may not also be an epitope for HLA-allele Y. Thus, epitope prediction takes as input both a peptide and an HLA allele, and returns the probability (or some score) reflecting how likely that pair is to be an epitope. Note that HLA alleles are encoded in a hierarchy, where extra digits are used to refer to more specific forms of the allele. For example, moving up the hierarchy from more specific to less specific, we have, A*020101, A*0201, and A02. In addition, many 4-digit alleles belong to a "supertype"—for example, A*0201 belongs to the A2 supertype.

Typically, a single classifier is trained and tested *for each HLA allele* (where the allele is defined with respect to one specific level of the hierarchy) (Buus et al., 2003) or for each HLA supertype (Larsen et al., 2005). These approaches have several shortcomings. One can build classifiers only for alleles with a large number of known epitopes or for alleles which fall in to one of the currently defined supertypes—a fairly strong restriction. Also, if one builds allele-specific or supertype-specific classifiers, then any information which could have been shared across somewhat similarly behaving alleles or supertypes is lost. Because sample sizes are usually extremely small, this shortcoming could be huge in some cases. With supertype classifiers, one is dependent upon the current definitions of supertypes, which has not been rigorously tested in a quantitative way. It may also be the case that some information contained in epitopes is very general, not specific to either alleles or supertypes. Thus, it would be desirable to *simultaneously leverage* epitope information from a number of sources when making epitope predictions:

1. Within specific HLA alleles (as available and appropriate)
2. Within specific HLA supertypes (as available and appropriate)
3. Across all epitopes, regardless of supertype or allele (as appropriate)

That is, in predicting whether a peptide is an epitope for a given HLA allele, we would like to use all information available to us, not just information about epitopes for this allele, but from information about epitopes for other alleles within this allele's supertype (if it has one), and from information about other

epitopes of any HLA type. Also, we would like to learn automatically when each type of information is appropriate, and to what degree, allowing us to combine them in a principled way for prediction.

The essence of how we achieve this goal is in the features we use, and is also related to the fact that we train on all HLA alleles and supertypes simultaneously with these features even though our model makes predictions on whether a peptide is an epitope for a specific HLA allele. In the simplest application to epitope prediction, a separate model would either be built for each HLA-allele, or for each supertype, and the features (inputs to the model) would be the amino acid sequence of the peptide, or some encoding of these, such as those discussed for example in Nielsen et al. (2003). Standard elaborations to this simple approach, in any domain, include using higher order moments of the data (e.g., pairwise statistics of neighboring amino acids) as features in addition to the features of single amino acids. While such higher-order statistics may improve epitope prediction, such experimentation is not the focus of our work. Instead, as mentioned above, we seek to leverage information across HLA alleles and supertypes, and do so by learning a single model for all HLA alleles using features of the form (1) position i has a particular amino acid or chemical property and the epitope's HLA allele is Y, which when used alone would be roughly equivalent to simultaneously building separate models for each HLA allele, as well as (2) position i has a particular amino acid or chemical property and the epitope's HLA has supertype Y, which helps leverage information across HLA alleles for a given supertype, and (3) position i has a particular amino acid or position i has an amino acid with a particular chemical property, which helps leverage information across all HLA alleles and supertypes. This leveraging approach can be applied to various classification models including logistic regression, support vector machines, and artificial neural networks. In our experiments, we show that our leveraging approach applied to logistic regression yields more accurate predictions than those generated from models learned on each supertype individually.

2. RELATED WORK

The general idea of leveraging has been described previously under the names "multitask learning" and "transfer learning" (Caruana, 1997). To our knowledge, the only published epitope prediction algorithm that might leverage information across alleles or supertypes is DistBoost (Yanover and Hertz, 2005), which could do so indirectly by learning a distance function across the entire space of epitopes (i.e., for all alleles or supertypes). However, they did not explicitly seek to leverage information in the way we have described, and therefore did not explicitly show that their algorithm does in fact leverage this type of information.

Other approaches to the problem of epitope prediction (or the slightly different problem of binding affinity prediction) include the use of weight matrices (sometimes called PSSMs—position-specific scoring matrices), whereby a probability distribution or score over amino acids at each position is used to make a prediction (Bhasin and Raghava, 2004a; Dong and Suie, 2005; Reche et al., 2004), artificial-neural-network approaches which are said to model amino acid position correlations in a fruitful way (Bhasin and Raghava, 2004a; Buus et al., 2003; Milik et al., 1998; Zhao et al., 2003), support vector machine (SVM) approaches (Bhasin and Raghava, 2004a, 2004b; Donnes and Elofsson 2002; Zhao et al., 2003), and decision trees (Zhao et al., 2003). In addition, there is the mostly hand-crafted SYFPEITHI classifier (Rammensee et al., 1999). The approach of Nielsen et al. (2003) also uses a Hidden Markov Model (HMM) whose output is used as feature for their neural network. In the recent approach of Larsen et al. (2005), they demonstrate that their binding affinity neural network approach combined with TAP transport efficiency predictors and proteasomal cleavage predictors does better than a non-integrated approach where the latter two pieces of information are not used.

Among the aforementioned papers, some (Bhasin and Raghava, 2004b; Buus et al., 2003; Dong and Suie, 2005; Donnes and Elofsson, 2002; Reche et al., 2004; Zhao et al., 2003) build classifiers for individual HLA alleles (or just a single HLA allele) using only data from each respective HLA class for training. Larsen et al. (2005) build classifiers for individual supertypes using only data from each respective supertype for training, while Nielsen et al. (2003) use some combination of the two, but never train on data outside of the respective allele or supertype. Furthermore, perhaps with the exception of PSSM-based approaches, our method is simpler to understand and to implement, yet outperforms PSSM-based methods, and also achieves better results than the most sophisticated methods.

3. LOGISTIC REGRESSION

Let y denote the binary variable (or class label) to be predicted and $\mathbf{x} = x_1, \ldots, x_k$ denote the binary (0/1) or continuous features to be used for prediction. In our case, y corresponds to whether or not a peptide–HLA pair is an epitope and the features correspond to 0/1 encodings of properties of the peptide–HLA pair. In this notation, the logistic regression model is

$$\log \frac{p(y|\mathbf{x})}{1 - p(y|\mathbf{x})} = w_0 + \sum_{i=1}^{k} w_i \cdot x_i, \tag{1}$$

where $\mathbf{w} = (w_0, \ldots, w_k)$ are the model parameters or weights. This follows from an alternate way of specifying the model,

$$p(y = 1|\mathbf{x}) = \left(1 + \exp\left(-w_0 - \sum_{i=1}^{k} w_i \cdot x_i\right)\right)^{-1} \tag{2}$$

$$p(y = 0|\mathbf{x}) = 1 - p(y = 1|\mathbf{x}). \tag{3}$$

Given a data set of cases $(y^1, \mathbf{x}^1), \ldots, (y^n, \mathbf{x}^n)$ that are independent and identically distributed given the model parameters, we learn the weights by assuming that they are mutually independent, each having a Gaussian prior $p(w_i|\sigma^2) = N(0, \sigma^2)$, and determining the weights that have the maximum a posteriori (MAP) probability. That is, we find the weights that maximize the quantity

$$\sum_{j=1}^{n} \log p(y^j|\mathbf{x}^j, \mathbf{w}) + \sum_{i=0}^{k} \log p(w_i|\sigma^2) \tag{4}$$

This optimization problem has a global maximum which can be found by a variety of techniques including gradient descent. We use the method (and code) of Goodman (2002), which he calls sequential conditional generalized iterative scaling. We tune σ^2 using ten-fold cross validation on the training data.

4. DATA AND METHODS

We used two data sets to evaluate our approach. The first, called MHCBN, contains selected 9mer–HLA and 9mer–supertype pairs from the MHCBN data repository. In this repository, both epitopes and non-epitopes are experimentally confirmed (Yanover and Hertz, 2005; Bhasin et al., 2003).

The second, called SYFPEITHI+LANL, includes all unique 9mer–HLA epitopes from the SYFPEITHI database (*www.syfpeithi.de*) in March 2004 and the Los Alamos HIV Database (*www.hiv.lanl.gov*) in December 2004. Examples not classified as human MHC class I (HLA-A, HLA-B, or HLA-C) were excluded, yielding 1287 and 339 positive examples of epitopes from SYFPEITHI and LANL, respectively. Neither SYFPEITHI nor LANL contains experimentally confirmed negatives, so we generated examples of non-epitope HLA–9mer pairs by randomly drawing from the distributions of HLAs and amino acids in the positive examples. The amino acid at each position in a 9mer was generated independently.[1] For each positive example, we generated 100 negative examples.

As the research in our lab focuses primarily on the prediction of HIV epitopes, we trained our models on both SYFPEITHI and LANL data, but then tested only on LANL data with appropriate cross validation. In particular, we used ten-fold cross validation where the training data of a given fold consisted of all

[1] In preliminary experiments, we found that, in contrast to the findings of Yanover and Hertz (2005) on MHCBN, the use of real negatives from a proprietary data source and the use of randomly generated negatives produced essentially the same results. Here, we report results for the randomly generated negatives, so that we may publish the data on which these results are based.

TABLE 1. MAPPING FROM HLA TO SUPERTYPE

Supertype	*HLAs*
A1	A01, A25, A26, A32, A36, A43, A80
A2	A02, A6802, A69
A3	A03, A11, A31, A33, A6801
A24	A23, A24, A30
B7	B07, B1508, B35, B51, B53, B54, B55, B56, B67, B78
B27	B14, B1503, B1509, B1510, B1518, B27, B38, B39, B48, B73
B44	B18, B37, B40, B41, B44, B45, B49, B50
B58	B1516, B1517, B57, B58
B62	B13, B13, B1501, B1502, B1506, B1512, B1513, B1514, B1519, B1521, B46, B52

Available at *www.hiv.lanl.gov/content/immunology/motif_scan/supertype.html*.

TABLE 2. FEATURE TYPES USED FOR PREDICTION

Feature type	*Description*
HLA	The HLA allele with 2 or 4 digit encoding; HLA=A02
Supertype (S)	The supertype of the HLA allele; S=A2
HLA $\wedge$ amino acid (AA)	Conjunction of HLA and AA; HLA=A02 and AA1=Ser
HLA $\wedge$ chemical property (CP)	Conjunction of HLA and CP; HLA=A02 and polar(AA1)
S $\wedge$ AA	Conjunction of S and AA; S=A2 and AA1=Ser
S $\wedge$ CP	Conjunction of S and CP; S=A2 and polar(AA1=Ser)
AA	Amino acid at a given position in the peptide; AA1=Ser
CP	Chemical property of amino acid at given position; polar(AA1)

Examples are shown for the peptide SLYNTVATL which is an epitope for HLA allele A*0201, which in turn belongs to the A2 supertype.

SYFPEITHI data and nine-tenths of the LANL data, and the test data consisted of one-tenth of the LANL data. If an epitope appeared in both SYFPEITHI and LANL, we treated it as if it were in LANL only. As mentioned, HLA alleles are encoded in a hierarchy. Because many examples in the SYFPEITHI and LANL databases have HLA alleles encoded only to two digits, we encoded all our examples with two-digit HLA alleles, except for the allele classes B15xx and A68xx, which have elements that belong to different supertypes. There are several supertype classifications; we used the one available from LANL shown in Table 1. The train–test splits of each fold are available at *ftp://ftp.research.microsoft.com/users/heckerma/recomb06*.

As discussed, we introduced a variety of feature types in an effort to leverage information across HLA alleles and supertypes. The types of features that we used are described in Table 2. In addition to features representing the presence or absence of amino acids at positions along the epitope, we included features representing the chemical properties of the amino acids in our LR models. For example, we used the chemical properties available at *www.geneinfinity.org/rastop/manual/aatable.htm*: cyclic, aliphatic, aromatic, hydrophobic, buried, large, medium, small, negative, positive, charged, and polar.

Using a large number of features in LR can lead to poor prediction unless some method for feature selection is used (Kohavi, 1995). In our experiments, we set the Z weights with the smallest magnitudes to zero, where Z was determined by optimizing the average log probability of prediction on a ten-fold cross validation of the training set. (We used these same cross-validation runs to tune σ^2.) In our largest model, which used all feature types and was trained on all of the data, this feature selection method chose 3,180 out of 23,852 features.[2]

[2]Many more than 23,852 features were possible, but only this many were warranted based on the training data (e.g., if amino acid Arg was never found in position 3, then no corresponding feature was created).

Finally, to evaluate prediction accuracy, we used ROC curves—in particular, plots of the false-positive rate (% non-epitopes identified as epitopes) versus the false-negative rate (% epitopes missed). We summarized the prediction accuracy for a given method using the area under the curve (AUC) of the ROC. To determine whether two methods are significantly different, for each distinct false-negative value, we determined corresponding false-positive values for the two methods, and applied the resulting pairs to a two-sided Wilcoxon matched-pairs signed-ranks test. We deemed a difference to be significant if its p-value (corrected for multiple tests when appropriate) was less than 0.05.

5. RESULTS

First, we examined whether LR with our features can leverage information about epitopes associated with a variety of supertypes and/or HLA alleles to help predict epitopes associated with different supertypes and/or alleles. To do so, for each supertype (including "none"), we compared the predictive accuracy of a leveraged model that was learned from all training examples with a non-leveraged or individual model that was trained only on epitopes (and non-epitopes) associated with that supertype. Our comparison used ten-fold cross validation, stratified by class label. We pooled the results across the ten folds before generating the ROC curves. In this case, pooling was justified because LR models produce calibrated probabilities. Figure 1 shows ROC curves for leveraged and individual models for each supertype. Leveraging helps significantly for two of the supertypes (A24 and B7),[3] and helps dramatically when predicting epitopes whose HLA alleles have no supertype. In two cases (B27 and B62), the AUC for predictions of the leveraged model is greater than that for non-leveraged model, but the differences are not significant.

To tease apart which features (global, HLA-conjunctive, or supertype-conjunctive) are contributing to the predictive performance, we selectively removed each of the conjunctive features in turn, and then together. Figure 2 shows the resulting aggregate (over all HLA types) ROC curves for this experiment. We again used ten-fold cross validation, stratified by class label and then pooled the results across the ten folds before generating the ROC curves. The use of only global features (no conjunctive features) results in terrible prediction accuracy (AUC $= 0.178$ compared to 0.0696, 0.0726, and 0.0862, with, for e.g., $p = 2.4\times10^{-34}$ in comparison to using "No Supertype")—not suprising given that we expect epitope prediction to be fairly HLA-specific, or at least supertype specific. Using any single type of conjunctive feature provides a big win, though the HLA-conjunctive features provide a slightly bigger win than the supertype-conjunctive features ($p = 4.25 \times 10^{-14}$), and use of both conjunctive features provides a small win over either conjunctive feature alone ($p = 3.6\times10^{-4}$ relative to "No Supertype," and $p = 6.3\times10^{-23}$ relative to "No HLA"). Thus we see that use of all conjunctive features produces the best predictive performance, and that HLA-specific features alone provide slighly more benefit that supertype-specific features alone. However, we note that these results are somewhat specific to the particular data set on hand. For example, a data set consisting of far more HLA alleles with no known supertype could increase the relative benefit of the HLA-conjunctive features, and a data set consisting of many members of each supertype could decrease the relative benefit of the HLA-conjunctive features. Nevertheless, with any data set, it is clearly beneficial to apply all conjunctive features and to then learn the best combination.

Lastly, we compared the predictions of our (leveraged) LR model with those of DistBoost. In their paper, Yanover and Hertz (2005) compared their approach to RANKPEP (PSSM), NetMHC (artificial neural network), and SVMHC (support-vector machine). Their comparison used a 70/30 train–test split of a subset of the MHCBN data set, and evaluated performance on A2 supertype epitopes. Yanover and Hertz found that DistBoost predicted significantly better than the other methods. Here, we compared DistBoost with LR on this same data and on the SYFPEITHI+LANL data,[4] in both cases using five-fold cross

[3]The p-value of 0.0267 for A2 is not significant after Bonferroni correction.

[4]After the publication of Yanover and Hertz (2005), the entry AYAKAAAAF–A02 was deleted from the MHCBN repository. We similarly deleted this entry from the MHCBN data set. The SYFPEITHI+LANL data set contained nine entries with unique HLA types. We deleted these entries as they could not be processed by DistBoost. In addition, we used only one negative example for every positive example to accommodate DistBoost's computational requirements, and used the feature encoding of Yanover and Hertz (2005) when training and testing with DistBoost. The train–test splits of each fold for both comparisons are available at *ftp://ftp.research.microsoft.com/users/heckerma/recomb06*.

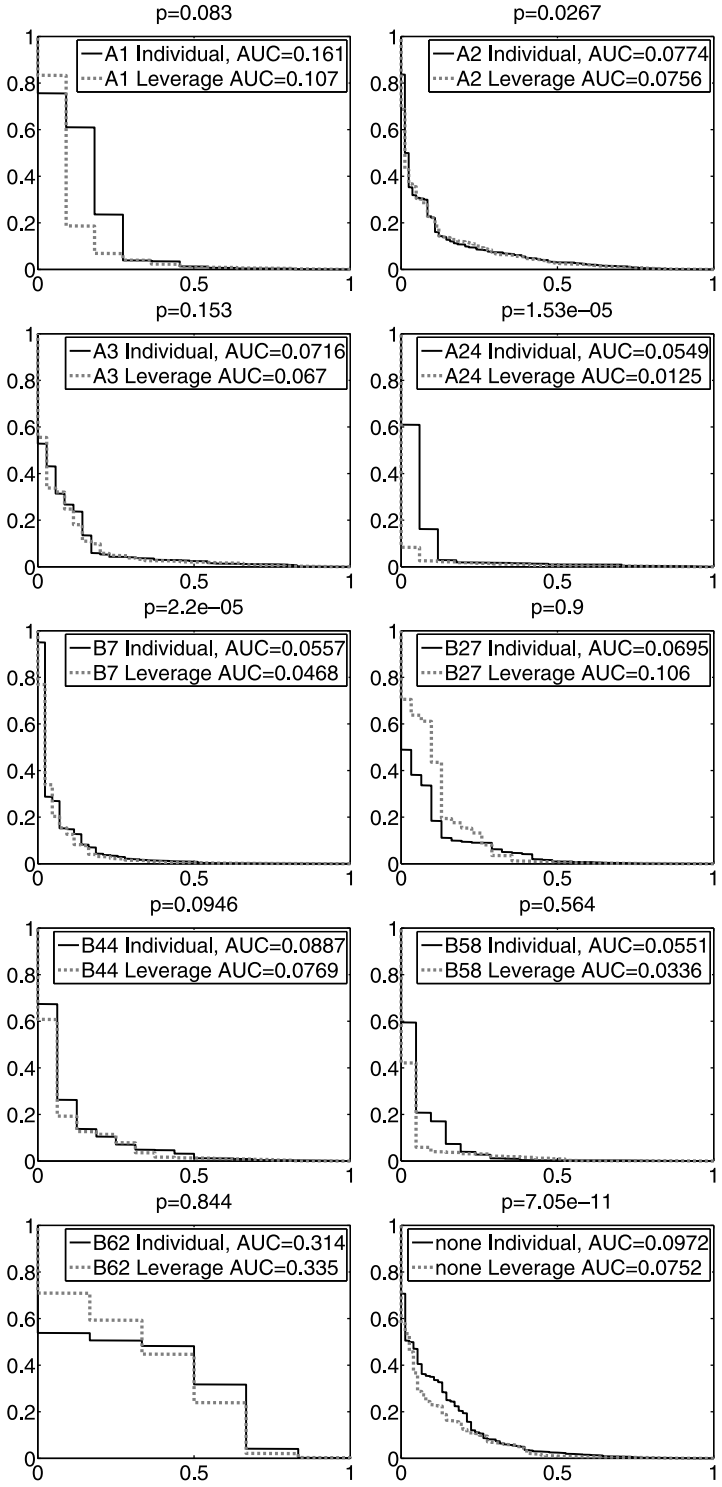

FIG. 1. ROC curves, AUCs, and *p*-values (not Bonferroni corrected) for leveraged and non-leveraged (individual) predictions of epitopes having alleles in each supertype (including "none," i.e., those not belonging to any supertype). ROC curves plot false-positive rate versus false-negative rate.

validation stratified by class label and HLA. We ran DistBoost for 30 boosting iterations on the MHCBN data set, and for 50 iterations on the larger SYFPEITHI+LANL data set. (We also tried 100 boosting iterations for each data set, with no substantial change in results.) The results, illustrated in Figures 3 and 4, indicate that the predictive accuracy of LR is better than that of DistBoost. Two-sided p-values computed from false-positive rates pooled across the five folds of the MHCBN and SYFPEITHI+LANL data are 1.8210e-08 and 5.1581e-29, respectively.

Finally, it is interesting to look at the learned features and their weights to see where leveraging is taking place. Table 3 contains a portion of a model trained on the full SYFPEITHI+LANL data set. The forty

TABLE 3. A PORTION OF A MODEL LEARNED FROM THE FULL SYFPEITHI+LANL DATA SET

Weight	*Feature*
−3.87821	large(AA1)
−3.01895	S=A1
2.8267	S=B27 and AA2=Arg
−2.61487	polar(AA1)
−2.48691	large(AA2)
−2.09559	HLA=A01
−1.83075	polar(AA2)
1.73488	S=A1 and polar(AA1)
1.71218	S=A1 and charged(AA1)
1.66352	S=B27 and positive(AA2)
−1.62407	charged(AA1)
1.47669	S=A24 and AA2=Tyr
−1.4628	aliphatic(AA3)
1.45694	negative(AA2)
1.44531	S=A1 and large(AA1)
−1.39833	AA1=Pro
1.35753	S=B44 and large(AA2)
−1.32388	buried(AA4)
1.31555	HLA=B27 and large(AA2)
1.29462	AA4=Trp
1.28076	HLA=B27 and AA2=Arg
1.27827	S=B44 and AA2=Glu
1.26313	HLA=A02 and AA2=Leu
−1.26253	medium(AA1)
1.24698	S=A1 and hydrophobic(AA3)
1.24487	S=B62 and AA2=Gln
1.22292	S=A24 and charged(AA1)
1.19599	S=A24 and positive(AA1)
1.18911	S=A1 and aliphatic(AA3)
−1.17646	charged(AA2)
1.16866	S=A3 and positive(AA1)
1.09196	S=B27 and large(AA2)
1.08261	HLA=A02 and large(AA1)
1.07628	S=B7 and AA2=Pro
−1.07365	S=B44 and hydrophobic(AA2)
1.04742	AA4=Pro
1.04397	S=none and large(AA1)
−1.0417	S=B27 and hydrophobic(AA2)
1.03173	AA9=Leu
1.02222	HLA=A02 and polar(AA1)

The forty features with the largest magnitude weights are shown. Positive weights increase the probability of being an epitope. Feature names are described in Table 2.

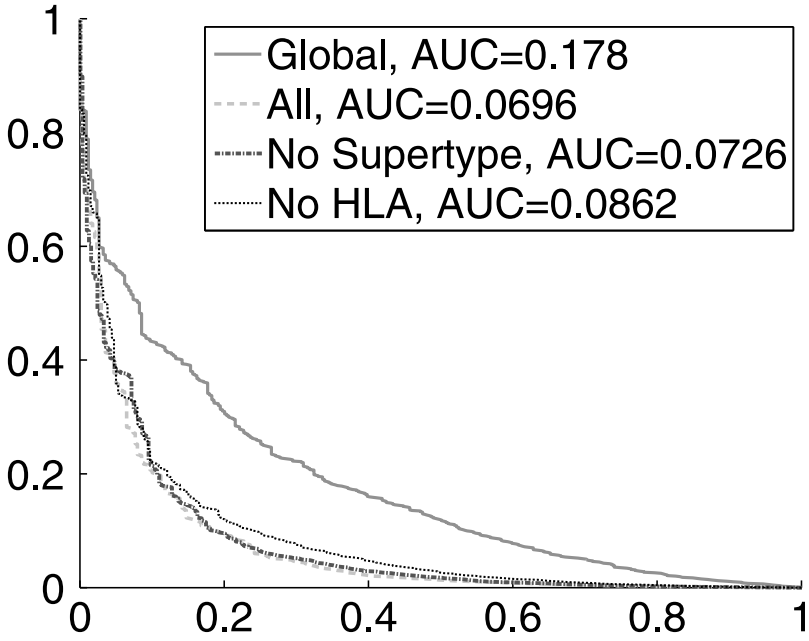

FIG. 2. ROC curves and AUCs, for, in the order of the figure legend, (i) no conjunctive features (i.e., global features), (ii) using all of global features, HLA-conjunctive features and supertype-conjunctive features, (iii) using global features and HLA-conjunctive features, and (iv) using global features and supertype-conjunctive features.

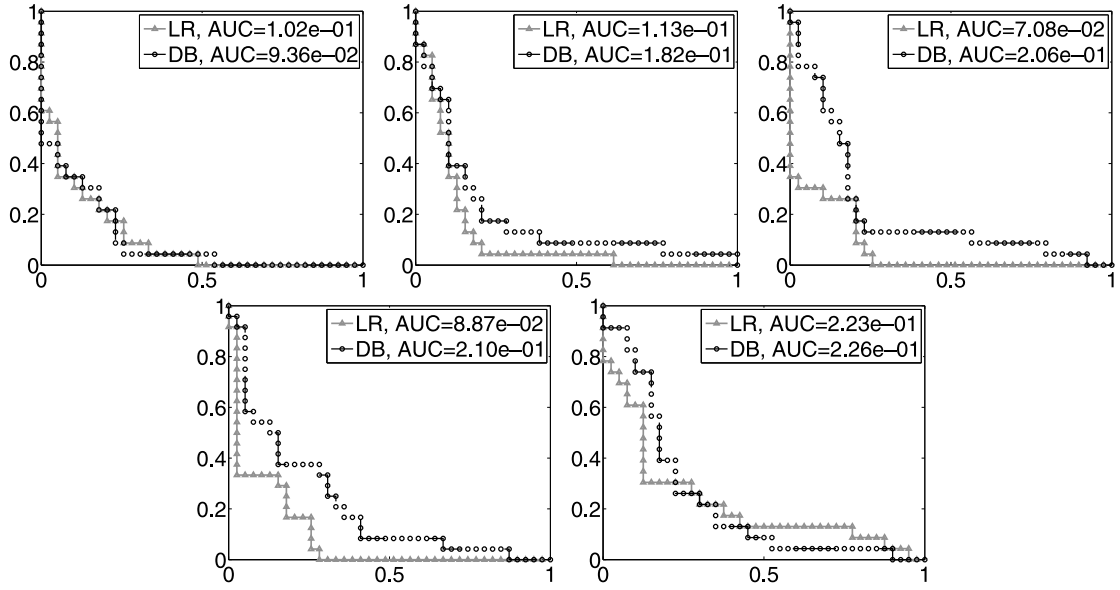

FIG. 3. ROC curves for LR and DistBoost applied to five-fold cross validation 9mer data from MHCBN. The two-sided *p*-value from false-positive rates pooled across the five folds is 1.8210e-08.

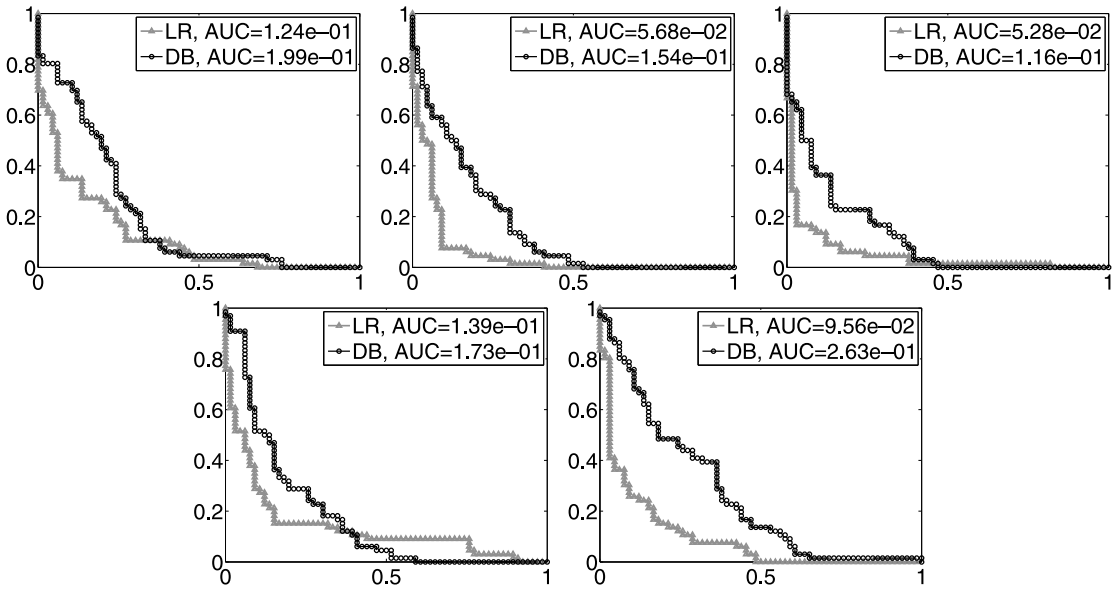

FIG. 4. ROC curves for LR and DistBoost applied to five-fold cross validation of the SYFPEITHI+LANL data. The two-sided *p*-value from false-positive rates pooled across the five folds is 5.1581e-29.

features with the largest magnitude weights are shown. Many of these strong features are general (e.g., large(AA1) and polar(AA1)) or contain conjunctions with supertypes (e.g., Supertype=A1 and polar(AA1)) and thereby facilitate leveraging.

6. DISCUSSION

We have presented a model for predicting HLA class I restricted CTL epitopes. Our model, which is based on logistic regression, is simple to implement and understand, is solved by finding a single global maximum, and is more accurate (to our knowledge) than the best published results. In addition, we have shown how to leverage information about epitopes having one allele or supertype to predict epitopes having different alleles or supertypes. We have shown that this leveraging can improve prediction of epitopes having HLA alleles with known supertypes, and dramatically increases our ability to predict epitopes having alleles which do not fall into any of the known supertypes.

Our next steps will be to build and evaluate LR predictors for HLA class I epitopes of lengths eight, ten, and eleven amino acids. In addition, rather than use a predefined set of supertypes, we plan to learn a set of (overlapping) supertypes that lead to accurate prediction. In particular, we plan to extend the LR model to include hidden variables that represent these new supertypes. Finally, we are looking at whether the inclusion of additional features such as distances between amino acids in the epitope and in the HLA molecule when the epitope and HLA molecule are in their minimum-energy configuration can improve prediction accuracy.

ACKNOWLEDGMENTS

We thank Vladimir Jojic for extracting and parsing the SYFPEITHI data set as well as Chen Yanover and Tomer Hertz for providing us with their data and software. Work done by J.L. was supported by an internship at Microsoft Research while away from her studies in the Department of Computer Science, University of Toronto.

REFERENCES

Bhasin, M., and Raghava, G. 2004a. Prediction of CTL epitopes using QM, SVM and ANN techniques. *Vaccine* 22, 3195–3204.

Bhasin, M., and Raghava, G.P.S. 2004b. SVM-based method for predicting HLA-DRB1*0401 binding peptides in an antigen sequence. *Bioinformatics* 20, 421–423.

Bhasin, M., Singh, H., and Raghava, G. 2003. MHCBN: a comprehensive database of MHC binding and non-binding peptides. *Bioinformatics* 19, 665–666.

Buus, S., Lauemoller, S., Worning, P., et al. 2003. Sensitive quantitative predictions of peptide-MHC binding by a "query by committee" artificial neural network approach. *Tissue Antigens* 62, 378–384.

Caruana, R. 1997. Multitask learning [PhD dissertation]. School of Computer Science, Carnegie Mellon University, Pittsburgh.

Dong, H.-L., and Suie, Y.-F. 2005. Prediction of HLA-A2-restricted CTL epitope specific to HCC by SYFPEITHI combined with polynomial method. *World J. Gastroenterol.* 2, 208–211.

Donnes, P., and Elofsson, A. 2002. Prediction of MHC class I binding. *BMC Bioinform.* 3, p. 25.

Goodman, J. 2002. Sequential conditional generalized iterative scaling. *ACL.* Proceedings of the 40th Annual Meeting of Association for Computational Linguistics, 2001, Philadelphia, PA, pg. 9–16. Association for Computational Linguistics, Morristown, NJ.

Goulder, P., Addo, M., Altfeld, M., et al. 2001. Rapid definition of five novel HLA-A*3002-restricted human immunodeficiency virus-specific cytotoxic T-lymphocyte epitopes by Elispot and intracellular cytokine staining assays. *J. Virol.* 75, 1339–1347.

Kohavi, R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proc. 14th Int. Joint Conf. Art. Intell.*, 1137–1145.

Larsen, M., Lundegaard, C., Lamberth, K., et al. 2005. An integrative approach to CTL epitope prediction: a combined algorithm integrating MHC class I binding, TAP transport efficiency, and proteasomal cleavage predictions. *Eur. J. Immunol.* 35, 2295–2303.

McMichael, A., and Hanke, T. 2002. The quest for an aids vaccine: is the $CD8^+$ T-cell approach feasible? *Nat. Rev.* 2, 283–291.

Milik, M., Sauer, D., Brunmark, A., et al. 1998. Application of an artificial neural network to predict specific class I MHC binding peptide sequences. *Nat. Biotechnol.* 16, 753–756.

Nielsen, M., Lundegaard, C., Worning, P., et al. 2003. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Sci.* 5, 1007–1017.

Parham, P. 2004. *The Immune System*. Garland Science Publishing, New York.

Platt, J. 1999. Probabilities for support vector machines, 61–74. In: *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA.

Rammensee, H., Bachmann, J., Emmerich, N., et al. 1999. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* 50, 213–219.

Reche, P., Glutting, J., Zhang, H., et al. 2004. Enhancement to the Rankpep resource for the prediction of peptide binding to MHC molecules using profiles. *Immunogenetics* 26, 405–419.

Yanover, C., and Hertz, T. 2005. Predicting protein-peptide binding affinity by learning peptide-peptide distance functions. *RECOMB 2005*, 456–471.

Zhao, Y., Pinilla, C., Valmori, D., et al. 2003. Application of support vector machines for T-cell epitopes prediction. *Bioinformatics* 19, 1978–1984.

Address reprint requests to:
Dr. David Heckerman
Microsoft Research
Redmond, WA 98052

E-mail: heckerma@microsoft.com