Published in final edited form as:
IEEE/ACM Trans Comput Biol Bioinform. 2011 ; 8(4): 1067–1079. doi:10.1109/TCBB.2010.94.

Predicting MHC-II binding affinity using multiple instance regression

Yasser EL-Manzalawy
Department of Systems and Computers Engineering, Al-Azhar University, Cairo, Egypt. yasser@azhar.edu.eg

Drena Dobbs
Department of Genetics, Development and Cell Biology, Bioinformatics and Computational Biology Graduate Program, Iowa State University Ames, IA 50010, USA. ddobbs@iastate.edu

Vasant Honavar [Senior Member, IEEE]
Artificial Intelligence Laboratory, Department of Computer Science, Bioinformatics and Computational Biology Graduate Program, Center for Computational Intelligence, Learning, and Discovery, Iowa State University Ames, IA 50010, USA. honavar@iastate.edu

Abstract

Reliably predicting the ability of antigen peptides to bind to major histocompatibility complex class II (MHC-II) molecules is an essential step in developing new vaccines. Uncovering the amino acid sequence correlates of the binding affinity of MHC-II binding peptides is important for understanding pathogenesis and immune response. The task of predicting MHC-II binding peptides is complicated by the significant variability in their length. Most existing computational methods for predicting MHC-II binding peptides focus on identifying a nine amino acids core region in each binding peptide. We formulate the problems of qualitatively and quantitatively predicting flexible length MHC-II peptides as multiple instance learning and multiple instance regression problems, respectively. Based on this formulation, we introduce MHCMIR, a novel method for predicting MHC-II binding affinity using multiple instance regression. We present results of experiments using several benchmark datasets that show that MHCMIR is competitive with the state-of-the-art methods for predicting MHC-II binding peptides. An online web server that implements the MHCMIR method for MHC-II binding affinity prediction is freely accessible at http://ailab.cs.iastate.edu/mhcmir.

Keywords

MHC-II peptide prediction; multiple instance learning; multiple instance regression

I. Introduction

T-cells, a major type of the immune system cells, play a central role in the cell-mediated immunity [1]. Cytotoxic T-cells attack cells that have certain foreign or abnormal molecules on their surfaces. They have also been implicated in transplant rejection. Helper T-cells, or CD4+ T-cells, coordinate immune responses by communicating with other cells. Once activated, they divide rapidly and secrete cytokines that regulate the immune response. T-cells are also targets of HIV infection, with the loss of CD4+ T-cells being associated with the appearance of AIDS symptoms. Regulatory T-cells are believed to be crucial for the maintenance of immunological tolerance. T-cells epitopes are short linear peptides that are generated by the cleavage of antigenic proteins. The identification of T-cell epitopes in protein sequences is important for understanding disease pathogenesis, for identifying potential autoantigens, and for designing vaccines and immune-based cancer therapies. Predicting whether a given peptide will bind to a specific major histocompatibility complex (MHC) molecule (and its binding affinity) is an important step in identifying potential T-cell epitopes. Consequently, predicting MHC binding peptides is an important and challenging task in immunoinformatics [2], [3].

There are two classes of MHC molecules: MHC class I (MHC-I) molecules that are characterized by short binding peptides, usually consisting of 9 amino acid residues; and MHC class II (MHC-II) molecules that bind to peptides of variable length. MHC-II binding peptides typically vary from 11 to 30 amino acids in length, although shorter and longer MHC-binding peptides are not entirely uncommon [4]. MHC-II molecules allow variable length peptides to bind because the binding groove of MHC-II molecule is open at both ends. However, it has been reported that a 9-mer core region is essential for MHC-II binding activity of peptides [4], [5]. Because the precise location of the 9-mer core region of the MHC-II binding peptide is unknown, predicting MHC-II binding peptides is more challenging than predicting MHC-I binding peptides.

The computational methods that are currently available for predicting MHC-II peptides can be grouped into two major categories:

- Quantitative MHC-II binding prediction methods that attempt to predict the binding affinities (e.g., IC50 values); Examples of such methods include PLS-ISC [6], MHCPred [7], SVRMHC [8], ARB [9], and NetMHCII [10].
- Qualitative MHC-II binding prediction methods that simply classify MHC peptides into binders and non-binders; Examples of such methods include: (i) methods that use a position weight matrix to model ungapped multiple sequence alignment of MHC binding peptides [10], [11], [12], [13], [14], or rely on Hidden Markov Models (HMMs) [15], [16]; (ii) supervised machine learning methods based on Artificial Neural Networks (ANN) [17], [18] or Support Vector Machines (SVMs) [19], [20], [21], [22]; and (iii) semi-supervised machine learning methods [23], [24].

Several MHC-II binding prediction methods focus on identifying a putative 9-mer MHC-II binding core region, e.g., based on the degree of match with a 9-mer MHC-II binding motif, typically constructed using one of the motif finding algorithms. For example, MEME [25], Gibbs sampling [26], matrix optimization techniques (MOTs) (Singh and Raghava, unpublished data), evolutionary algorithms [27], Monte Carlo (MC) search [28], and linear programming [29] form the basis of MHC-II binding peptide prediction methods RankPEP [11], Gibbs [13], HLA-DR4Pred [20], MOEA [14], NetMHCII [10], and LP [23], respectively. The success of these MHC-II prediction methods in identifying MHC-II peptides relies on the effectiveness of the corresponding motif-finding methods in recognizing the motif that characterizes the 9-mer core of MHC-II binding peptides.

An inherent limitation of MHC-II peptide prediction methods that focus on identifying 9-mer cores is their inability to exploit potentially useful predictive information that may be available outside the 9-mer core region. For example, Chang et al. [30] have shown that incorporating peptide length as one of the inputs improves the performance of the predictor (relative to one that uses only the features derived from the 9-mer core) in the case of several MHC-II alleles; Nielsen et al. [10] have demonstrated that including peptide flanking residues among inputs improves the performance of their SMM-align method on 11 out of 14 MHC-II allele-specific datasets.

Recently, two MHC-II binding peptide prediction methods [21], [22] that do not rely on the pre-identification of the 9-mer binding cores in the training data have been proposed. Both methods use the entire sequences of MHC-II peptides (as opposed to only the 9-mer cores) for training MHC-II binding peptide predictors. The first method [21] maps a variable length peptide into a fixed length feature vector obtained from sequence-derived structural and physicochemical properties of the peptide. The second method [22] uses a sequence kernel that defines the pairwise similarity of variable-length peptides as the average score of all possible local alignments between the corresponding amino acid sequences. Both these representations of peptides can be used to train predictors that classify a peptide of any length as an MHC-II binder or a non-binder (i.e., qualitative MHC-II predictors), or predict its MHC-II binding affinity (i.e., quantitative MHC-II predictors). However, these two approaches do not help identify the binding core of the query peptide.

Against this background, the main contributions of this paper to the current state-of-the-art in predicting MHC-II peptides are as follows:

(i) Novel multiple instance learning (MIL) and multiple instance regression (MIR) formulations of the flexible length MHC-II binding peptide prediction problem and the MHC-II peptide affinity prediction problem, respectively. The multiple instance representation of flexible length peptides encodes a peptide sequence, regardless of its length, by a *bag* of 9-mer subsequences. The label associated with each bag could be either binary label indicating whether the corresponding peptide is an MHC-II binder or not or could be numeric label indicating the corresponding binding affinity of the peptide. An attractive feature of the proposed method (that is also shared by some of the recently developed MHC-II binding peptide prediction methods, e.g., [23], [31]) is that it does not require the 9-mer cores in each binding peptide to be identified prior to training the predictor. The 9-mer binding cores are identified by the learning algorithm based on the features of MHC-II binders and non-binders so as to optimize the predictive performance of the learned model.

(ii) MILESreg, an adaptation of MILES [32] for multiple instance regression on bags of amino acid sequences.

(iii) MHCMIR, a novel method for predicting the binding affinity of flexible length MHC-II peptides using MILESreg. The performance of MHCMIR estimated using statistical cross-validation on a benchmark dataset, covering 16 HLA and mouse MHC-II alleles, as well as on independent test data, shows that MHCMIR is competitive with the state-of-the-art methods for predicting MHC-II binding peptides on a majority of MHC-II alleles. These results demonstrate the utility and promise of multiple instance representation of peptides in advancing the current state-of-the-art in MHC-II binding peptide prediction. An implementation of MHCMIR as an online web server for predicting MHC-II binding affinity is freely accessible at http://ailab.cs.iastate.edu/mhcmir.

II. Multiple Instance Learning

The multiple instance learning (MIL) problem, first introduced by Dietterich et al. [33] was motivated by a challenging classification task in drug discovery where the goal is to determine whether or not a given molecule is likely to bind to a desired protein binding site. In this task, each molecule can adopt multiple shapes (conformations) as a consequence of rotation of some internal bonds. A good drug candidate is one that has one or more conformations that bind tightly to the desired binding site on a target protein whereas a poor drug candidate is one that has no conformations that bind tightly to the desired binding site on the target protein. A multiple instance learning formulation of this problem [33] involves representing each candidate molecule by a *bag* of instances, with each instance in the bag representing a unique conformation assumed by the molecule. Under the so-called *standard* *multiple instance learning assumption*, a molecule (i.e., the corresponding bag of conformations) is labeled positive if and only if at least one of the conformations in the bag binds tightly to the desired binding site on the target protein; Otherwise, it is labeled negative. More generally, a bag is labeled positive if it contains at least one positive instance, and negative otherwise. During classification, the MIL classifier is given a bag of instances to be assigned a positive or negative label based on the instances in the bag. What makes the MIL problem challenging is the fact that the learning algorithm has access to the contents of, and the label assigned to, each bag; but has no knowledge of the specific instance(s) in a positively labeled bag that are responsible for the positive label.

In the standard (single instance) supervised classifier learning scenario, typically, each instance (input to the classifier) is represented by an ordered tuple of attribute values in the instance space $I = D_1 \times D_2 \times \ldots \times D_n$, where D_i is the domain of the i^{th} attribute. The output of the classifier is a class label drawn from a set C of mutually exclusive classes. A training example is a labeled instance in the form $\langle X_i, c(X_i) \rangle$ where $X_i \in I$ and $c : I \rightarrow C$ is unknown function that assigns to an instance X_i its corresponding class label $c(X_i)$. For simplicity we consider only the binary classification problem in which $C = \{-1, 1\}$. Given a collection of training examples, $E = \{\langle X_1, c(X_1)\rangle, \ldots, \langle X_n, c(X_n)\rangle\}$, the goal of the (single instance) learner is to learn a function c^* that approximates c as well as possible (as measured by some pre-specified performance criterion, e.g., accuracy of classification).

The MIL problem involves training a classifier to label *bags* of instances (as opposed to individual instances as is usually the case in the standard supervised learning scenario). Let $B = \{B_1, B_2, \ldots, B_m\}$ be a collection of bags. Let $B_i = \{X_{i1}, X_{i2}, \ldots X_{iki}\}$ denote a bag of k_i instances ($k_i \geq 1$). The set of MIC training examples, E_{MI}, is a collection of ordered pairs $\langle B_i, f(B_i)\rangle$ where f is unknown function that assigns to each bag B_i a class label $f(B_i) \in \{-1, 1\}$. Under the standard multiple instance learning assumption [33], $f(B_i) = -1$ iff $\forall j \, \varepsilon \, \{1 \ldots k_i\}$, $c(X_{ij}) = -1$; and $f(B_i) = 1$ iff $\exists j \, \varepsilon \, \{1 \ldots \}$, such that $c(X_{ij}) = 1$. Given E_{MI}, a collection of MI training examples, the goal of an MIC learner is to learn as good an approximation of the function f as possible (as measured by some pre-specified performance measure e.g., accuracy of classification of bags).

Dietterich et al. [33] proposed a solution to the MIL problem under the standard MIL assumption using a hypothesis space of axis-parallel rectangles (see Figures 1 and 2). Figure 1 (adapted from [33]) shows a schematic diagram of the MIL problem wherein instances are represented as points in a two dimensional Euclidean instance space. Instances that belong to the same bag are shown using the same shape. Unfilled shapes represent instances that belong to the positively labeled bags; filled shapes represent instances that belong to the negatively labeled bags. An axis parallel rectangle is used to classify bags as follows: a bag is assigned a positive label if at least one of its instances is contained within the rectangle; and a negative label otherwise. In this setting, given a set of labeled bags, the goal of the MIL algorithm is to identify an axis parallel rectangle that includes at least one unfilled point of each shape (i.e., at least one positively labeled instance from each positively labeled bag) and does not include any filled points (i.e., instances from negatively labeled bags). Such a solution is shown in Figure 2.

Subsequently, many solutions to the MIL problem and its variants have been investigated in the literature. Ramon and De Raedt [34] introduced a variant of the back-propagation algorithm for training a neural network for multiple instance classification problem. Wang and Zucker [35] proposed variants of the k-nearest neighbor (k-NN) algorithm. Maron and Lozano-Perez [36] introduced the *diverse density* (DD) framework for solving multiple instance classifier learning problems. The basic idea behind the DD method is to locate a point in the feature space that is close to at least one instance from every positive bag and as far away as possible from instances in the negative bags. Zhang and Goldman proposed EM-DD [37] which improves on DD by using Expectation Maximization (EM). The difficulty of MIL comes from the ambiguity of not knowing which of the instances in a bag is most likely to be responsible for its positive label. EM-DD models the mapping of instances to labels assigned to the bag using a set of *hidden variables*, which are estimated using the EM. EM-DD starts with an initial guess of the solution (obtained using original DD algorithm), and refines the guess by applying EM. Andrews et al. [38] and Gartner et al. [39] have proposed adaptations of support vector machines that involve changing the objective function or the kernel function to suit the multiple instance classification problem. Ray and Craven [40] compared several multiple instance classifier learning algorithms as well as their standard supervised learning counterparts. Scott et al. [41] introduced a generalization of the multiple instance learning model in which all of the instances in a bag are used to determine its label. Tao et al. [42] have explored kernel functions for the generalized multiple instance learning problem.

MIL algorithms have been used, with varying degrees of success, on a number of practical applications including: content-based image retrieval (CBIR) [43], [44] in which each image is viewed as a bag of objects (image regions) and an image is assigned a label based on the presence or absence of specific objects; web page classification [45] in which each web page is modeled by a bag of pages that it links to, and is labeled positive based on the user's interest in at least one of the pages that a given page links to; and computer-aided diagnosis [46] in which each medical case is modeled by a bag of medical images (e.g., CT scans, X-ray, MRI etc) and is labeled positively if at least one of these medical images indicate malignant tumors and lesions.

The multiple instance regression (MIR) problem is a generalization of the MIL problem where each bag is labeled with a real number (as opposed to a discrete class label). Several MIR algorithms have been reported in the literature including [37], [47], [48].

A. MIL formulation of the MHC-II binding peptide prediction problem

We now proceed to introduce an MIL formulation of the variable length MHC-II binding peptide prediction problem.

Recall that a 9-mer core region is believed to be essential for MHC-II binding [4], [5]. We represent each variable length MHC-II peptide sequence by a bag of all 9-mer subsequences extracted from it. Under the *standard MIL assumption*, we assign a positive label to a bag of 9-mers extracted from an MHC-II binding peptide; and a negative label to a bag of 9-mers extracted from a non MHC-II binding peptide. Figure 3 shows an example of an MHC-II binding peptide and its mapping into a bag of 9-mer subsequences. It should be noted that labels are associated with bags of 9-mers, and not individual 9-mers. Consequently, in preparing the training data, we do not need to know which of the 9-mers in a bag (if any) is a binding core.

The problem of learning to predict the MHC-II binding affinities of flexible length peptides can be formulated as a multiple instance regression problem in a manner similar to that described above for the classification setting, simply by mapping each peptide into a bag of 9-mers and substituting the class labels with the measured real-valued binding affinities for each peptide.

In summary, both qualitative and quantitative predictions of the MHC-II binding activity of peptides can be obtained using predictive models based on the multiple instance formulations of the corresponding classification and regression problems (respectively). The resulting problems can be solved using the multiple instance learning algorithms or multiple instance regression algorithms as appropriate. In this paper, we focus on the quantitative prediction of the binding activity of MHC-II peptides using a multiple instance regression algorithm.

III. Materials and Methods

A. Cross-validation dataset

We used the IEDB benchmark dataset, introduced by Nielsen et al. [10], in our experiments. The dataset consists of peptides along with their IC50 binding affinities for 14 HLA-DR and three H2-IA alleles (hereafter referred to as IEDB dataset for short). Details of the IEDB benchmark dataset are summarized in Table I. Because each peptide is labeled with its binding affinity (IC50) value, peptides were categorized into binders and non-binders using a binding affinity threshold of 500 nM [10]. To avoid overly optimistic estimates of the performance of MHC-II binding peptide prediction methods, it is important to ensure that the peptide sequences used to evaluate the performance of the predictor do not share a high degree of sequence overlap (or similarity) with peptide sequences in the training set used to train the predictor. Nielsen et al. [10] have provided a partitioning of each IEDB allele dataset into five subsets so as to minimize the degree of sequence overlap between any pair of subsets. Following [14], from this data, we excluded the DRB3-0101 MHC-II allele dataset in our experiments because of its highly skewed distribution (only 3 binders as opposed to 99 non-binders). We used the data for the rest of the MHC-II alleles in our 5-fold cross-validation experiments. That is, for each MHC-II allele, in each of the 5 runs of a cross-validation experiments, 4 of the 5 subsets of the allele-specific data were used for training the predictor and the remaining subset was used as the test set for evaluating the performance of the trained predictor. The predictions on the 5 disjoint test sets used in the 5 cross-validation runs were then combined to obtain a single estimate of performance.

B. Independent validation datasets

We assessed the performance of the predictors trained using MHCMIR method on IEDB allele datasets [10] by measuring their performance on three independent validation datasets: i) IDS-Wang, a dataset published by Wang et al. [49], which is a comprehensive dataset of previously unpublished 10,017 MHC-II binding affinities spanning 114 proteins and covering 14 HLA alleles and two mouse MHC-II alleles (See Table V); ii) IDSLin, a dataset published by Lin et al. [50], which is a set of 103 peptides extracted from four antigens and covering seven HLA alleles (DRB1*0101, 0301, 0401, 0701, 1101, 1301, and 1501); iii) IDS-Nielsen, a binding core identification dataset published by Nielsen et al. [51], which is a set of 15 MHC-peptide complexes extracted from Protein Data Bank (PDB) database [52]. For each peptide in these structures, the 9-mer binding core was manually identified by determining which peptide residue is pound in the P1 pocket of the MHC-II molecule.

C. MHCMIR method

In order to explore the feasibility of predicting MHC-II binding activity of peptides based on the proposed multiple instance regression formulation, we developed MHCMIR, a novel method for predicting the binding affinity of MHC-II peptides using multiple instance regression. Given a dataset of MHC-II peptides where each peptide is labeled with its experimentally determined binding affinity (IC50 value), MHCMIR maps each peptide to its corresponding bag of 9-mers and uses the data in its multiple instance representation to train a multiple instance regression model. The learned multiple instance regression model can be used to predict the affinity of any query peptide by providing as input to the model the bag of 9-mers representation of the query peptide sequence.

In this study, we chose to adapt MILES (multiple instance learning via embedded selection) [32], an algorithm for training multiple instance classifiers, to work in the regression setting. MILES maps each bag of instances into a meta instance constructed by applying an Euclidean distance based similarity measure to instances within each bag. Then, a 1-norm SVM classifier [53] is trained on the resulting dataset of meta instances. The competitive performance of MILES, and its low computational cost of training are some of its main advantages relative to other MIL algorithms [32].

Adapting the MILES algorithm for training a multiple instance classifier into a multiple instance regression algorithm is rather straightforward. All we need to do is to replace the 1-norm SVM classifier by a support vector regression (SVR) model [54]. Because in our application, the bags to be labeled comprise 9-mers over the amino-acid alphabet, we replaced the Euclidean distance used in MILES for transforming a bag of instances into a meta instance by a distance function that is customized for calculating the distance between amino acid sequences. This distance function is based on the BLOSUM62 amino acid substitution matrix [55].

The pseudocode shown in Algorithm 1 summarizes MILESreg, our proposed multiple instance regression algorithm. The function $dist(s1, s2)$ computes the distance between two 9-mers, $s1$ and $s2$. Note that $BLOSUM62(aa1, aa2)$ is the corresponding BLOSUM62 matrix entry for the amino acids $aa1$ and $aa2$ and $s[i]$ denotes the amino acid in the i^{th} position in the sequence s.

Predicting the label of a test bag B_i is performed in two steps. First, B_i is mapped into a meta instance using the set of training instances C and the procedure described in lines 3 to 6 in the pseudocode. Then, a predicted real value is assigned to the meta-instance using the learned support vector regression model.

D. Performance evaluation

We used the area under Receiver Operating Characteristic (ROC) curve for a predictor as a measure of its performance on a classification task. The ROC curve is obtained by plotting the true positive rate as a function of the false positive rate or, equivalently, sensitivity versus (1-specificity) as the discrimination threshold of the binary classifier is varied. Each point on the ROC curve corresponds to the performance of the classifier at a specific choice of the *classification threshold*, i.e., at a particular choice of the tradeoff between true positive rate and false positive rate. The area under ROC curve (AUC) is a useful summary statistic for comparing two ROC curves. The AUC corresponds to the probability that a randomly chosen positive example will be ranked higher than a randomly chosen negative example by the classifier when the numeric output (before applying the threshold) or score assigned by the classifier to an input sample is used to rank the input sample. The higher the score assigned to a sample, the higher the rank. An ideal classifier will have an AUC = 1, whereas a classifier that assigns labels at random will have an AUC = 0.5, and any classifier with performance that is better than random will have an AUC value that lies between 0,5 and 1. Swets [56] has suggested that the values AUC ≥ 0.9 indicates excellent, 0.9 > AUC ≥ 0.8 good, 0.8 > AUC ≥ 0.7 marginal, and AUC < 0.7 poor predictions.

In IEDB, IDS-Lin, and IDS-Wang datasets, peptides are labeled with their experimentally reported binding affinities (e.g., IC50 values). However, estimating the AUC for the predictors in a classification setting requires the binding affinities to be mapped to a binary class label (MHC-II binder versus non-binder) for each peptide. Different choices of the cutoff on the binding affinity values can yield different classifications for the same peptide. Several different cutoffs have been used in previous studies [9], [10], [49], [50]. Recently, Wang et al. [49] have examined the effect of different choices of the cutoff (in the range between 50 nM and 5000 nM) on the estimated performance of MHC-II prediction methods as measured by the AUC. They concluded that the estimated AUC is relatively independent of the specific choice of the cutoff over this range. In our experiments, in order to ensure fair comparison between the various methods, we labeled the peptides as MHC-II binders versus non-binders using the same cutoffs on binding affinity as those used by the developers of each of the respective benchmark datasets. Specifically, we used 500 nM cutoff for IEDB and 1000 nM cutoff for IDS-Lin and IDS-Wang datasets.

IV. Results

A. Cross-validation evaluation of MHCMIR

We compared the predictive performance of MHCMIR with that of several MHC-II binding peptide prediction methods reported in the literature: Gibbs sampler [13], TEPITOPE [57], SVRMHC [8], MHCPred [7], ARB [9], NetMCHII (also called SMM-align) [10], and MOEA [14]. Because most reports of MHC-II binding activity prediction methods in the literature focus on qualitative prediction of MHC-II binding activity, although MHCMIR is able to produce both quantitative and qualitative predictions of MHC-II binding activity (the latter by comparing the predicted binding affinity value with a threshold), our comparisons focus on qualitative predictions of MHC-II binding activity. Specifically, we compared the estimated area under ROC curve (AUC) [56] for the different methods. In the case of MHCMIR, Gibbs sampler, NetMHCII [10], and MOEA [14] the performance estimates were obtained using 5-fold cross-validation on the partitioning of each MHC-II allele dataset into 5 subsets, ensuring minimal sequence overlap between the different subsets provided by Nielsen et al. [10]. Because the codes for the SVRMHC, MHCPred, and ARB methods are not readily available, estimates of the performance of these methods were obtained by submitting the data to the online web servers that implement the respective methods (using the default parameter setting for each server). As noted in [10], the reported performance of ARB method should be interpreted with caution because the ARB method has been trained on data from IEDB database [58], which, because of the overlap between the training and test data, gives it an unfair advantage over other methods.

Table II compares the predictive performance, in terms of AUC, of the different MHC-II peptide prediction methods. “-” indicates information that is unavailable either because the online server does not provide predictions on the corresponding allele (e.g., SVRMHC, MHCPred, and ARB on a number of allele datasets) or because the data was not reported in the published studies of the predictor (e.g., detailed results of Gibbs method on the three mouse allele datasets were not provided in [10]).

In addition to the AUC, Table III compares the performance of the different MHC-II peptide prediction methods as estimated by the Pearson's correlation coefficient [59] between the predicted and actual labels. MOEA has been excluded from this comparison because its performance has been reported using only AUC [14]. Overall, these results show that the performance of MHCMIR is competitive with that of the state-of-the-art methods for predicting MHC-II binding peptides. However, no single method appears to consistently outperform all others. This observation underscores the practical utility of consensus methods for predicting MHC-II binding peptides [49].

B. Evaluation of MHCMIR predictive performance on validation test sets

We used three independent validation datasets for evaluating MHC-II peptide prediction methods. The first dataset, published by Wang et al. [49], which we call IDS-Wang, is a comprehensive data for previously unpublished MHC-II binding affinities covering 16 human and mouse MHC-II allele-specific datasets. The second dataset, published by Lin et al. [50], which we call IDS-Lin, is a set of 103 peptides extracted from four antigens and covering seven HLA alleles (DRB1*0101, 0301, 0401, 0701, 1101, 1301, and 1501). We also considered a third dataset, published by Nielsen et al. [51], which we call IDS-Nielsen, of 15 MHC-peptide complexes extracted from Protein Data Bank (PDB) database [52] to assess performance of MHCMIR in identifying the 9-mer binding cores. Here, we compare the predictive performance of MHCMIR with that of several MHC-II peptide prediction servers on IDS-Wang, IDS-Lin, and IDS-Nielsen, following the procedures described in by Wang et al. [49], Lin et al. [50] and Nielsen et al. [51], respectively.

Table V compares the AUC scores of MHCMIR with that of several MHC-II peptide prediction servers on the validation dataset IDS-Wang. If we compare the servers using the average AUC across all available MHC-II allele datasets following the procedure used by Wang et al. [49], MHCMIR, SMM-align [10] and PROPRED [12] have the best average AUC value (0.73). If we rank each server according to the number of datasets on which it has the best reported performance divided by the number of datasets available to the server, then MHCMIR and PROPRED have the best ranks (5/14 and 4/11, respectively), followed by SMM-align (4/15), ARB (2/15), and RANKPEP (1/14).

C. Statistical analysis of cross-validation results

In comparing two classifiers, statistical tests can be employed to determine whether the difference in performance between the two classifiers is significant or not. For comparing multiple classifiers on multiple datasets, we followed a procedure that has recently been recommended by Demšar [60] which involves comparing the average rank of the classifiers across the different datasets.

The statistical analysis of the performance comparisons was limited to NetMHCII, MOEA, and MHCMIR methods because these are only the methods with reported performance (AUC) on each of the allele datasets. First, the different classifiers are ranked on the basis of their observed performance on each dataset (see Table IV). Then we used the Friedman test to determine whether the measured average ranks are significantly different from the mean rank under the null hypothesis. We found that at 0.05 level of significance the null hypothesis could not be rejected. Hence, we concluded that the reported performances of the three methods are not significantly different.

Figure 4 shows the predictive performance (in terms of AUC) of MHCMIR on the validation dataset IDS-Lin. The results were obtained by submitting the four antigen sequences to the MHCMIR server using the default peptide length setting of 15. On each submitted protein sequence, MHCMIR returns a prediction for each 15-mer in the submitted sequence. To compare MHCMIR predictions with experimental data in IDS-Lin, which includes peptides ranging from 15 to 19 amino acids in length, we used two strategies that have been used for a similar purpose in [50]: the predicted binding affinity of the target variable length peptide was set to (i) the maximum score over the overlapping 15-mer peptides spanning the length of a target peptide; ii) the average score of the overlapping 15-mer peptides spanning the length of a target peptide. Figure 4 shows that the two strategies produces almost identical results for all the alleles except DRB*1101 and DRB*1501 where the first method yields slightly higher AUC values. Because the results of Lin et al. [50] are unavailable in a tabular form, it is not possible to directly compare the 21 servers evaluated in [50] with MHCMIR. However, it is reasonable to infer from Figure 1 in [50], that the performance of MHCMIR on each allele-specific dataset is highly competitive with the best performing servers among the 21 servers compared by Lin et al. [50].

Finally, we assessed the performance of MHCMIR in identifying the 9-mer binding cores on IDS-Nielsen, a dataset of 15 MHC-peptide complexes [51]. Each query peptide was submitted to MHCMIR server. The MHC-II allele option was selected to reflect the target allele (See Table VI). To predict 9-mer cores, the peptide length was set to 9 amino acids. Therefore, MHCMIR server returned a prediction score for each 9-mer sub-peptide. If we consider the highest scoring 9-mer as the predicted MHC-II binding core, then MHCMIR is found to correctly identify the MHC-II binding cores in 9 out of the 15 MHC-II binding peptides (as compared to NetMHCIIpan which correctly identified 14 binding cores) (See Table VI). However, if we relax the criterion for correct identification of a binding core so as to consider a binding core as correctly identified if it is one of the top 2 highest scoring 9-mers, then MHCMIR is found to correctly identify the binding cores of the entire set of 15 MHC-II binding peptides. A possible explanation for this result is that MILESreg, the MIR algorithm used to train the MHC-II binding peptide predictors in MHCMIR, operates under the assumption that *one or more* 9-mer sub-peptides of the target peptide contribute the binding affinity of the peptide.

D. Reduced multiple instance representation of MHC-II peptides

We have defined the multiple instance representation of a peptide p of length n as simply a bag of $(n - 9 + 1)$ 9-mer sub-peptides. However, there is growing evidence [4], [61] that the residue in the first position of the 9-mer binding core has to be hydrophobic amino acid (Y, F, W, I, V, L or M). We can exploit this information to reduce the number of 9-mers per bag by eliminating the 9-mers that do not contain a hydrophobic residue at their first positions (P1). Figure 5 compares the AUC values of MILESreg predictors on IEDB dataset when each peptide is represented by a bag of its constituent 9-mers and by a bag of a subset of its 9-mers that have a hydrophobic amino acid residue in their first positions. The results show that MILESreg predictors using the latter (reduced) multiple instance representation of MHC-II peptides have better AUC values than their counterparts that use the original multiple instance representations on 7 allele datasets (DRB1*0404, 0701, 0802, 0901, 1302, DRB5*0101, and H2-IAd) out of the 16 datasets.

E. Incorporating peptide flanking residues into MHC-II peptide multiple instance representation

Nielsen et al. [10] has noted that amino acid composition of peptide flanking residues (PFR) plays some role in stabilizing the peptide:MHC-II complex. Based on this observation, Nielsen et al. [10] showed that encoding the amino acid composition of the PFR as additional inputs to their SMM-align predictors slightly but consistently improves its predictive performance.

One way of incorporating PFR into our multiple instance representation of MHC-II peptides is to simply represent each peptide as a bag of 10, 11, or 12 -mers extracted from it. However, this representation does not necessarily reflect the widely held belief that the binding cores of MHC-II binding peptides are 9 amino acids long [4], [5]. However, from a machine learning perspective, whether incorporating the PFR data into the classifier input does indeed improve the accuracy of MHC-II binding peptide predictors is an empirical question. As such, this question can be answered by comparing the performance of classifiers that incorporate PFR data as input to the predictors with those that do not.

Table VII compares the AUC values of MILESreg on the IEDB dataset when each peptide is represented as a bag of 9-mers (i.e., not incorporating PFR) as opposed to when each peptide is represented using 10, 11, and 12 -mers (i.e., incorporating PFR). Our results show that, in the case of MILESreg, incorporating PFR does not yield improvements over of the baseline performance of the original bag of 9-mers representation. There are two possible explanations for the discrepancy between our results and those of Nielsen et al. [10] with respect to the benefits of incorporating PFR into the representation of MHC-II peptides: (i)

MILESreg assumes that one or more of the 9-mers extracted from a peptide contribute to its binding affinity whereas SMM-align [10] assumes a single 9-mer determines the MHC-II binding affinity of the peptide; (ii) SMM-align uses an encoding of PFR in terms of their amino acid composition whereas in our experiments we used an encoding of PFR in the form of bags of 10, 11, and 12-mer sequences. It would therefore be interesting to experiment with several alternative MIL algorithms using different encodings of PFR.

V. Summary and Discussion

Recently, several comparative studies [49], [50] suggest that the performance of MHC-II peptide prediction method is far from optimal and there is significant room for improvement in the performance of the state-of-the-art MHC-II binding peptide predictors. There are two primary directions to explore in terms of improving the performance of MHC-II binding peptide predictors: (i) compiling more representative experimentally well-characterized datasets for training and evaluating the performance of the predictors and (ii) exploring alternative data representations and machine learning methods. The primary focus of this study was on exploring the utility of a multiple instance representation of peptides for predicting MHC-II binding peptides. Specifically, we have introduced a novel formulation of the problem of learning to predict variable length MHC-II binding peptides as an instance of a multiple instance learning problem. The proposed method shares an attractive feature of some of the recently developed MHC-II binding peptide prediction methods [23], [31] in that it does not require that the 9-mer cores in each binding peptide be identified prior to training the predictor. The 9-mer binding cores are identified by the learning algorithm based on the features of MHC-II binders and non-binders so as to optimize the predictive performance of the learned model.

We have introduced MHCMIR, a multiple instance regression based method for predicting the binding affinity of variable length MHC-II peptides. MHCMIR utilizes MILESreg, our adaptation of MILES algorithm [32] for training multiple instance classifiers, for performing multiple instance regression where the input to the predictor is a bag of peptides. The results of our experiments using statistical cross-validation on benchmark datasets as well as additional independent test sets have shown that the proposed method although it does not substantially outperform the state-of-the-art methods, is quite competitive with the best performing methods that are currently available for predicting MHC-II binding peptides. These results demonstrate the utility and promise of multiple instance representation of peptides in advancing the current state-of-the art in MHC-II binding peptide prediction. We have made our implementation of MHCMIR freely available to the scientific community in the form of an online web server for predicting the binding affinity of MHC-II peptides. The server can be accessed at http://ailab.cs.iastate.edu/mhcmir.

The multiple instance representation of MHC-II peptides combined with a MIL or MIR method provides a general 2-component framework for developing a broad class of MHC-II prediction methods: (i) We can adapt the multiple instance representation of MHC-II peptide to incorporate different assumptions (e.g., the utility of PFR in predicting MHC-II binding peptides); (ii) We can choose any of the MIL and MIR algorithms available for training predictors using the multiple instance representation of peptides.

Current literature offers three broad classes of approaches to MIL or MIR learning based on different assumptions regarding the relation between the label assigned to a bag and the labels of the instances contained in that bag.

- Witness-based MIL or MIR methods [33], [36], [37], [38], [47], [62] which search for a single representative instance (witness) within each bag. Existing MHC-II prediction methods that search for a single 9-mer within each peptide [8], [9], [10],

[11], [13], [19], [14] are essentially instances of the *witness-based* MIL or MIR methods.

- Generalized MIL or MIR methods that operate under the assumption that all instances within a bag contribute the bag label [39], [40], [63]. Two recently proposed SVM-based MHC-II binding peptide prediction methods [21] and [22] which train SVM classifiers using the entire peptide sequence can be seen as variants of the *generalized* MIC learning methods. These two SVM-based qualitative MHC-II binding peptide prediction methods can be easily adapted to yield quantitative MHC-II predictions by replacing the SVM classifiers with support vector regression (SVR) models.
- Generalized MIL and MIR methods which operate under the assumption that only a subset of the instances within a bag contribute the bag label [32], [64]. The iterative approach for predicting MHC-II peptides [23] can be seen as an exemplar of this class of MIL and MIR methods.

In summary, our results have demonstrated the utility of multiple instance representation of peptides in both qualitative (i.e., MHC-II binder versus non-binder) as well as quantitative (i.e., binding affinity) prediction of MHC-II peptides. Our formulation of flexible length qualitative and quantitative MHC-II binding peptide prediction as multiple instance learning and multiple instance regression problems respectively has opened up the possibility of adapting a broad range of multiple instance methods for classification and regression in this setting.

Acknowledgments

This work was supported in part by a doctoral fellowship from the Egyptian Government to Yasser EL-Manzalawy, a grant from the National Institutes of Health (GM066387) to Vasant Honavar and Drena Dobbs, and a post-doctoral fellowship to Yasser EL-Manzalawy from the Iowa State University Center for Computational Intelligence, Learning, and Discovery.

Biographies

Yasser EL-Manzalawy received his Ph.D. in Computer Science in 2008 from Iowa State University. He is currently an assistant professor in Department of Systems and Computers Engineering, Al-Azhar University, Egypt. His research interests include bioinformatics, computational immunology, and machine learning.

Drena Dobbs received her Ph.D. in Molecular Biology from the University of Oregon in 1983 and additional training as an NIH Postdoctoral Fellow in Molecular Biology at the University of California, Berkeley. She is currently a professor of Genetics, Development and Cell Biology, and of Bioinformatics and Computational Biology at Iowa State University. Her research interests include prediction and validation of ligand binding residues in proteins (for protein, DNA, RNA and small molecules), rational design of zinc finger DNA binding proteins, and analysis of regulatory RNA-protein interactions in viruses. She has published over 80 refereed research articles on these topics during 1983–2010.

Vasant Honavar received his Ph.D. in Computer Science and Cognitive Science in 1990 from the University of Wisconsin (Madison). He joined the faculty at Iowa State University in 1990 where he is a professor of Computer Science and of Bioinformatics and Computational Biology. Honavar's research interests include machine learning (learning predictive models from sequence data, graph-structured data, relational data, multi-modal data), data mining (scalable approaches to building predictive models from autonomous, distributed, semantically disparate data sources), bioinformatics and computational molecular and systems biology (computational analysis, modeling, comparative analyses and prediction of protein-DNA, protein-RNA, protein-protein interfaces and macro-molecular interaction networks), information integration (ontology-based and probabilistic methods), knowledge representation (federated ontologies, privacy-preserving reasoning, epistemic description logics, representing and reasoning with qualitative preferences, logic-based methods for automated service composition, substitution, and adaptation), and health informatics. He has published over 200 refereed research articles on these topics during 1990–2010. Honavar is a senior member of IEEE and ACM, and a member of AAAI and ISCB.

References

[1]. Janeway, C.; Travers, P., et al. Immunobiology: The Immune System in Health and Disease. 6th ed.. Garland Pub; 2004.

[2]. Korber B, LaBute M, Yusim K. Immunoinformatics Comes of Age. PLoS Computational Biology. 2006; 2(no. 6):e71. [PubMed: 16846250]

[3]. Gowthaman U, Agrewala J. In Silico Tools for Predicting Peptides Binding to HLA-Class II Molecules: More Confusion than Conclusion. J Proteome Res. 2008; 7(no. 1):154–63. [PubMed: 18034454]

[4]. Rammensee H, Friede T, Stevanović S. MHC ligands and peptide motifs: first listing. Immunogenetics. 1995; 41(no. 4):178–228. [PubMed: 7890324]

[5]. Madden D. The three-dimensional structure of peptide-MHC complexes. Annual Review Immunology. 1995; 13:587–622.

[6]. Doytchinova I, Flower D. Towards the in silico identification of class II restricted T-cell epitopes: a partial least squares iterative self-consistent algorithm for affinity prediction. 2003:2263–2270.

[7]. Hattotuwagama C, Guan P, Doytchinova I, Zygouri C, Flower D. Quantitative online prediction of peptide binding to the major histo-compatibility complex. Journal of Molecular Graphics and Modelling. 2004; 22(no. 3):195–207. [PubMed: 14629978]

[8]. Liu W, Meng X, Xu Q, Flower D, Li T. Quantitative prediction of mouse class I MHC peptide binding affinity using support vector machine regression (SVR) models. BMC Bioinformatics. 2006; 7(no. 1):182. [PubMed: 16579851]

[9]. Bui H, Sidney J, Peters B, Sathiamurthy M, Sinichi A, Purton K, Mothé B, Chisari F, Watkins D, Sette A. Automated generation and evaluation of specific MHC binding predictive tools: ARB matrix applications. Immunogenetics. 2005; 57(no. 5):304–314. [PubMed: 15868141]

[10]. Nielsen M, Lundegaard C, Lund O. Prediction of MHC class II binding affinity using SMM-align, a novel stabilization matrix alignment method. BMC Bioinformatics. 2007; 8:238. [PubMed: 17608956]

[11]. Reche P, Glutting J, Zhang H, Reinherz E. Enhancement to the RANKPEP resource for the prediction of peptide binding to MHC molecules using profiles. Immunogenetics. 2004; 56(no. 6):405–419. [PubMed: 15349703]

[12]. Singh H, Raghava G. ProPred: prediction of HLA-DR binding sites. Bioinformatics. 2001; 17(no. 12):1236–1237. [PubMed: 11751237]

[13]. Nielsen M, Lundegaard C, Worning P, Sylvester-Hvid C, Lamberth K, Buus S, Brunak S, Lund O. Improved prediction of MHC class I and II epitopes using a novel Gibbs sampling approach. Bioinformatics. 2004; 20:1388–97. [PubMed: 14962912]

[14]. Rajapakse M, Schmidt B, Feng L, Brusic V. Predicting peptides binding to MHC class II molecules using multi-objective evolutionary algorithms. BMC Bioinformatics. 2007; 8(no. 1): 459. [PubMed: 18031584]

[15]. Mamitsuka H. Predicting peptides that bind to MHC molecules using supervised learning of Hidden Markov Models. PROTEINS: Structure, Function, and Genetics. 1998; 33:460–474.

[16]. Noguchi H, Kato R, Hanai T, Matsubara Y, Honda H, Brusic V, Kobayashi T. Hidden Markov Model-based prediction of antigenic peptides that interact with MHC class II molecules. Journal of Bioscience and Bioengineering. 2002; 94(no. 3):264–270. [PubMed: 16233301]

[17]. Nielsen M, Lundegaard C, Worning P, Lauemøller S, Lamberth K, Buus S, Brunak S, Lund O. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. Protein Science. 2003; 12:1007–1017. [PubMed: 12717023]

[18]. Buus S, Lauemoller S, Worning P, Kesmir C, Frimurer T, Corbet S, Fomsgaard A, Hilden J, Holm A, Brunak S. Sensitive quantitative predictions of peptide-MHC binding by a'Query by Committee' artificial neural network approach. Tissue Antigens. 2003; 62(no. 5):378–384. [PubMed: 14617044]

[19]. Donnes P, Kohlbacher O. SVMHC: a server for prediction of MHC-binding peptides. Nucleic Acids Research. 2006; 34(no. Web Server issue):W194. [PubMed: 16844990]

[20]. Bhasin M, Raghava G. SVM based method for predicting HLADRB1 0401 binding peptides in an antigen sequence. Bioinformatics. 2004; 20:3.

[21]. Cui J, Han L, Lin H, Zhang H, Tang Z, Zheng C, Cao Z, Chen Y. Prediction of MHC-binding peptides of flexible lengths from sequence-derived structural and physicochemical properties. Mol Immunol. 2006

[22]. Salomon J, Flower D. Predicting Class II MHC-Peptide binding: a kernel based approach using similarity scores. BMC Bioinformatics. 2006; 7(no. 1):501. [PubMed: 17105666]

[23]. Murugan N, Dai Y. Prediction of MHC class II binding peptides based on an iterative learning model. Immunome Research. 2005; 1(no. 1):6. [PubMed: 16351712]

[24]. Hertz T, Yanover C. PepDist: A New Framework for Protein-Peptide Binding Prediction based on Learning Peptide Distance Functions. BMC Bioinformatics. 2006; 7:S1–S3.

[25]. Bailey T, Elkan C. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. Machine Learning. 1995; 21(no. 1):51–80.

[26]. Lawrence C, Altschul S, Boguski M, Liu J, Neuwald A, Wootton J. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. Science. 1993; 262(no. 5131):208–214. [PubMed: 8211139]

[27]. Fonseca, C.; Fleming, P. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. Fifth International Conference on Genetic Algorithms; 1993. p. 416-423.

[28]. Metropolis N, Rosenbluth A, Rosenbluth M, Teller A, Teller E. Equation of State Calculations by Fast Computing Machines. The Journal of Chemical Physics. 2004; 21:1087.

[29]. Bennett K, Mangasarian O. Robust linear programming discrimination of two linearly inseparable sets. Optimization Methods and Software. 1992; 1(no. 1):23–34.

[30]. Chang S, Ghosh D, Kirschner D, Linderman J. Peptide length-based prediction of peptide-MHC class II binding. Bioinformatics. 2006; 22(no. 22):2761. [PubMed: 17000752]

[31]. Nielsen M, Lund O. NN-align. An artificial neural network-based alignment algorithm for MHC class II peptide binding prediction. BMC bioinformatics. 2009; 10(no. 1):296. [PubMed: 19765293]

[32]. Chen Y, Bi J, Wang J. MILES: Multiple-instance learning via embedded instance selection. IEEE Trans Pattern Anal Mach Intell. 2006; 28(no. 12):1931–1947. [PubMed: 17108368]

[33]. Dietterich RH, Lathrop TG, Lozano-Perez T. Solving the multiple-instance problem with axis parallel rectangles. Artificial Intelligence. 1997; 89(1–2):31–71.

[34]. Ramon, J.; De Raedt, L. Multi instance neural networks. ICML-2000 Workshop on Attribute-Value and Relational Learning; 2000.

[35]. Wang, J.; Zucker, JD. Solving the multiple-instance problem: a lazy learning approach. 17th International Conference on Machine Learning; 2000. p. 1119-1125.

[36]. Maron O, Lozano-Perez T. A framework for multiple-instance learning. Advances in Neural Information Processing Systems. 1998; 10:570–576.

[37]. Zhang Q, Goldman SA. Em-dd: An improved multiple-instance learning technique. Neural Information Processing Systems. 2001; 14

[38]. Andrews S, Tsochantaridis I, Hofmann T. Support vector machines for multiple-instance learning. Advances in Neural Information Processing Systems. 2002; 15

[39]. Gartner, T.; Flach, P.; Kowalczyk, A.; Smola, A. Multi-instance kernels. 19th International Conference on Machine Learning; 2002. p. 179-186.

[40]. Ray, S.; Craven, M. Supervised versus multiple instance learning: An empirical comparison. Twentieth-Second International Conference on Machine Learning; 2005. p. 697-704.

[41]. Scott S, Zhang J, Brown J. On Generalized Multiple-Instance Learning. International Journal of Computational Intelligence and Applications. 2005; 5(no. 1):21–35.

[42]. Tao Q, Scott S, Vinodchandran N, Osugi T, Mueller B. Kernels for Generalized Multiple-Instance Learning. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2007; 27

[43]. Maron, O.; Ratan, A. Multiple-Instance Learning for Natural Scene Classification. Fifteenth International Conference on Machine Learning table of contents; 1998. p. 341-349.

[44]. Zhang, Q.; Goldman, S.; Yu, W.; Fritts, J. Content-Based Image Retrieval Using Multiple-Instance Learning. Nineteenth International Conference on Machine Learning table of contents; 2002. p. 682-689.

[45]. Zhou Z, Jiang K, Li M. Multi-Instance Learning Based Web Mining. Applied Intelligence. 2005; 22(no. 2):135–147.

[46]. Fung, G.; Dundar, M.; Krishnapuram, B.; Rao, R. Multiple Instance Learning for Computer Aided Diagnosis. Advances in Neural Information Processing Systems: Proceedings of the 2006 Conference; 2007.

[47]. Ray, S.; Page, D. Multiple instance regression. Eighteenth International Conference on Machine Learning; 2001. p. 425-432.

[48]. Goldman S, Scott S. Multiple-Instance Learning of Real-Valued Geometric Patterns. Annals of Mathematics and Artificial Intelligence. 2003; 39(no. 3):259–290.

[49]. Wang P, Sidney J, Dow C, Mothé B, Sette A, Peters B. A Systematic Assessment of MHC Class II Peptide Binding Predictions and Evaluation of a Consensus Approach. PLoS Computational Biology. 2008; 4(no. 4)

[50]. Lin H, Zhang G, Tongchusak S, Reinherz E, Brusic V. Evaluation of MHC-II peptide binding prediction servers: applications for vaccine research. BMC bioinformatics. 2008; 9:S22. [PubMed: 19091022]

[51]. Nielsen M, Lundegaard C, Blicher T, Peters B, Sette A, Justesen S, Buus S, Lund O. Quantitative predictions of peptide binding to any HLA-DR molecule of known sequence: NetMHCIIpan. PLoS Computational Biology. 2008; 4(no. 7)

[52]. Berman H, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov I, Bourne P. The Protein Data Bank. Nucleic Acids Research. 2000; 28(no. 1):235–242. [PubMed: 10592235]

[53]. Zhu J, Kosset S, Hastie T, Tibshirani R. 1-norm Support Vector Machines. Advances in neural information processing systems. 2004

[54]. Shevade S, Keerthi S, Bhattacharyya C, Murthy K. Improvements to the SMO Algorithm for SVM Regression. IEEE Transactions on Neural Networks. 2000; 11(no. 5):1189.

[55]. Henikoff S, Henikoff J. Amino Acid Substitution Matrices from Protein Blocks. Proceedings of the National Academy of Sciences of the United States of America. 1992; 89(no. 22):10 915–10 919.

[56]. Swets J. Measuring the accuracy of diagnostic systems. Science. 1988; 240(no. 4857):1285–1293. [PubMed: 3287615]

[57]. Sturniolo T, Bono E, Ding J, Raddrizzani L, Tuereci O, Sahin U, Braxenthaler M, Gallazzi F, Protti M, Sinigaglia F, et al. Generation of tissue-specific and promiscuous HLA ligand databases using DNA microarrays and virtual HLA class II matrices. Nature Biotechnology. 1999; 17:555–561.

[58]. Peters B, Sidney J, Bourne P, Bui H, Buus S, Doh G, Fleri W, Kronenberg M, Kubo R, Lund O, et al. The Immune Epitope Database and Analysis Resource: From Vision to Blueprint. PLoS Biology. 2005; 3(no. 3)

[59]. Urdan, T. Statistics In Plain English. Lawrence Erlbaum Associates; 2005.

[60]. Demšar J. Statistical Comparisons of Classifiers over Multiple Data Sets. The Journal of Machine Learning Research. 2006; 7:1–30.

[61]. Brusic V, Rudy G, Honeyman G, Hammer J, Harrison L. Prediction of MHC class II-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics* (*Oxford, England*). 1998; 14(no. 2):121.

[62]. Mangasarian O, Wild E. Multiple instance classification via successive linear programming. Data Mining Institute Technical Report 05-02. 2005

[63]. Zhou Z, Zhang M. Solving multi-instance problems with classifier ensemble based on constructive clustering. Knowledge and Information Systems. 2007; 11(no. 2):155–170.

[64]. Weidmann, N.; Frank, E.; Pfahringer, B. A two-level learning method for generalized multi-instance problems. European Conference on Machine Learning; Springer; 2003. p. 468-479.

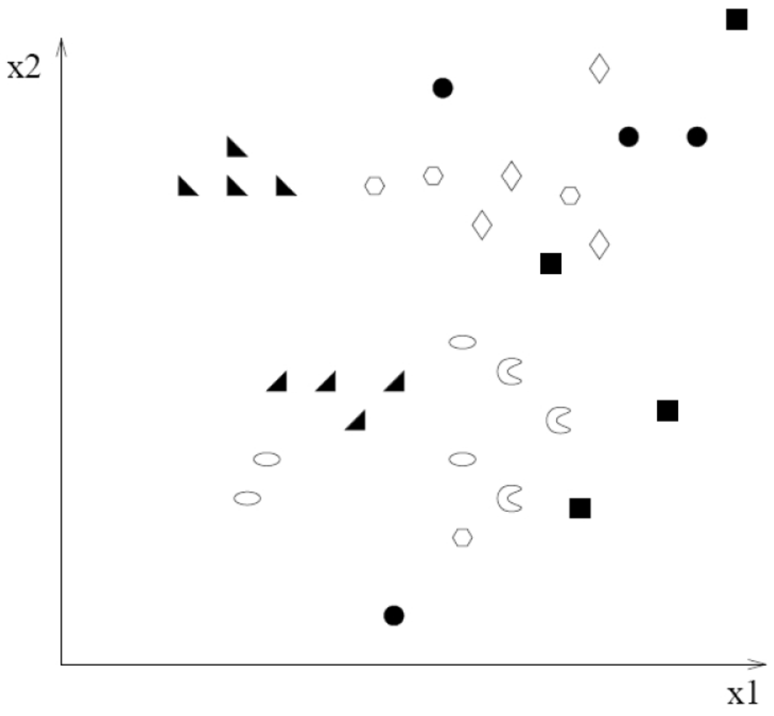

Fig. 1.

A multiple instance classifier learning problem. Unfilled shapes represent instances from positively labeled bags; filled shapes represent instances from negatively labeled bags. Instances extracted from the same bag are shown using the same shape. This figure is adapted from Figure 14 in [33].

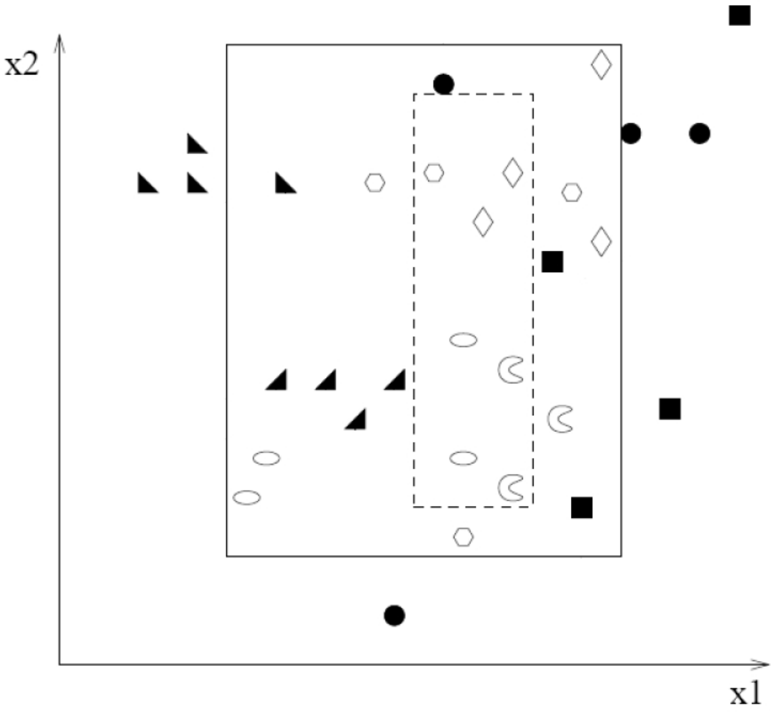

Fig. 2.

Solving the MIL problem using axis parallel rectangles (APR). The solid rectangle represents the initial solution, a rectangle that covers all instances that belong to positively labeled bags. The dashed rectangle represents the final APR solution, a rectangle that covers at least one positive instance from each positively labeled bag and no instances from negatively labeled bags. This figure is adapted from Figure 15 in [33].

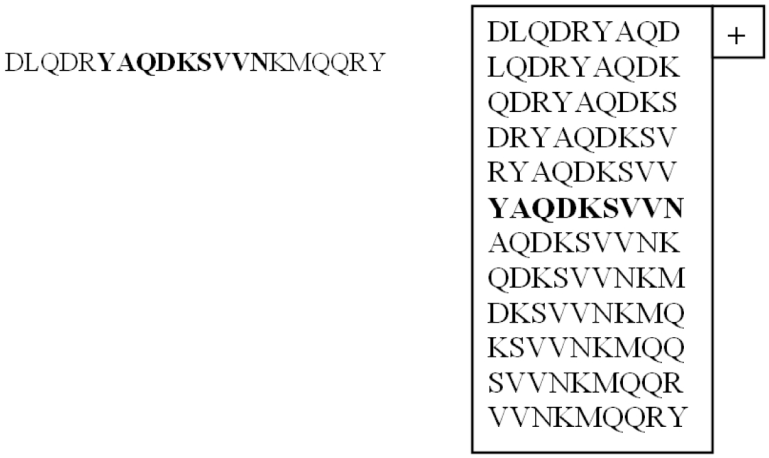

Fig. 3.

An example of an MHC-II binding peptide and its corresponding multiple instance representation. Bold subsequence indicates the 9-mer binding core. Mapping the peptide sequence into a bag does not require the identification of the 9-mer binding core because no labels are associated with the instances of the bag.

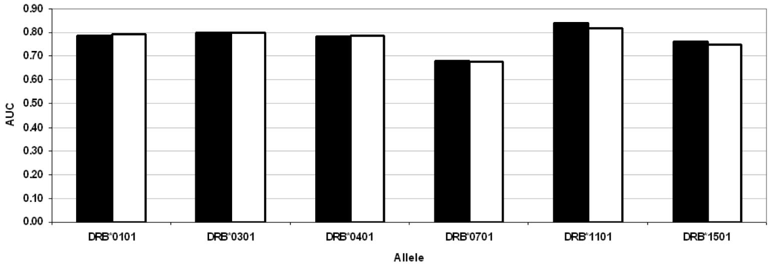

Fig. 4.

Performance of MHCMIR on IDS-Lin dataset [50]. Black bars for maximum 15-mer scores and white bars for average 15-mer scores.

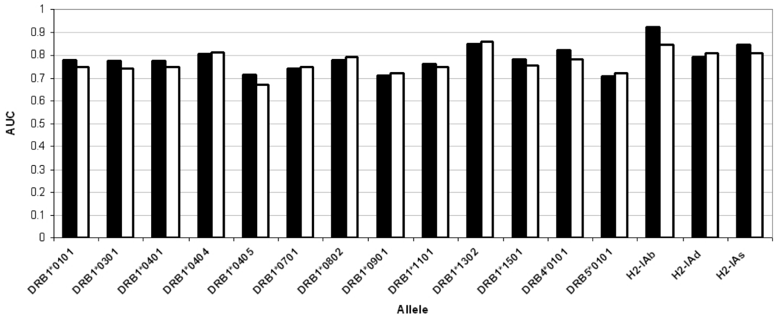

Fig. 5.

AUC values of MILESreg predictors on IEDB dataset when each peptide is represented as a bag of all 9-mers (black bars) and as a bag of only 9-mers with hydrophobic amino acid at P1 position (white bars), respectively.

TABLE I

Summary of the IEDB benchmark dataset [10]. Binding peptides were identified using an IC50 binding threshold of 500 nM.

Dataset	Binders	Non-binders
DRB1-0101	920	283
DRB1-0301	65	409
DRB1-0401	209	248
DRB1-0404	74	94
DRB1-0405	88	83
DRB1-0701	125	185
DRB1-0802	58	116
DRB1-0901	47	70
DRB1-1101	95	264
DRB1-1302	101	78
DRB1-1501	188	177
DRB3-0101	3	99
DRB4-0101	74	107
DRB5-0101	112	231
H2-IAb	43	33
H2-IAd	56	286
H2-IAs	35	91

TABLE II

Comparison of AUC values for the different MCH-II prediction methods on the IEDB benchmark dataset. Results for Gibbs, TEPITOPE, SVRMHC, MHCPred, ARB, and NetMHCII are taken from [10]. Results of MOEA are obtained from [14]. The results of the best-performing method are highlighted in bold.

Dataset	Gibbs	TEPITOPE	SVRMHC	MHCpred	ARB	NetMHCII	MOEA	MHCMIR
DRB1-0101	0.676	0.647	0.623	0.565	0.716	0.716	0.651	**0.778**
DRB1-0301	0.722	0.734	-	-	**0.799**	0.765	0.778	0.761
DRB1-0401	0.759	0.754	0.739	0.606	0.737	0.758	0.725	**0.760**
DRB1-0404	0.743	**0.829**	-	-	0.788	0.785	0.786	0.794
DRB1-0405	0.724	**0.790**	0.701	-	0.724	0.735	0.756	0.721
DRB1-0701	0.695	0.768	-	0.647	0.749	**0.787**	0.735	0.754
DRB1-0802	0.721	0.769	-	-	**0.803**	0.756	0.773	0.772
DRB1-0901	0.734	-	-	-	0.711	**0.775**	0.712	0.664
DRB1-1101	0.715	0.7150	-	-	0.727	0.734	**0.759**	0.734
DRB1-1302	0.716	0.720	-	-	**0.917**	0.818	0.820	0.852
DRB1-1501	0.672	0.726	0.730	-	**0.792**	0.736	0.743	0.774
DRB4-0101	0.742	-	-	-	0.800	0.736	0.759	**0.801**
DRB5-0101	0.618	0.653	0.649	-	**0.677**	0.664	0.660	0.675
H2-IAb	-	-	-	-	0.662	0.908	**0.919**	0.894
H2-IAd	-	-	-	-	0.819	0.818	**0.855**	0.775
H2-IAs	-	-	-	-	-	**0.898**	0.889	0.771

"-" indicates performance estimates that are currently unavailable (See text for details).

TABLE III

Comparison of Pearson's correlation coefficient values for the different MCH-II prediction methods on the IEDB benchmark dataset. Results for Gibbs, TEPITOPE, SVRMHC, MHCPred, ARB, and NetMHCII are obtained from [10]. The results of the best-performing method are highlighted in bold.

Dataset	Gibbs	TEPITOPE	SVRMHC	MHCpred	ARB	NetMHCIIMH	MHCMIR
DRB1-0101	0.260	0.333	0.213	0.146	0.376	0.413	**0.505**
DRB1-0301	0.453	0.227	-	-	0.506	0.466	**0.520**
DRB1-0401	0.482	**0.508**	0.461	0.176	0.434	0.499	0.488
DRB1-0404	0.433	**0.609**	-	-	0.529	0.481	0.476
DRB1-0405	0.428	**0.542**	0.409	-	0.420	0.417	0.444
DRB1-0701	0.353	0.460	-	0.309	0.410	**0.531**	0.432
DRB1-0802	0.375	0.472	-	-	**0.517**	0.461	0.457
DRB1-0901	0.398	-	-	-	0.440	**0.487**	0.343
DRB1-1101	0.385	0.382	-	-	0.421	0.426	**0.455**
DRB1-1302	0.400	0.411	-	-	**0.763**	0.594	0.624
DRB1-1501	0.305	0.453	0.481	-	**0.561**	0.461	0.535
DRB4-0101	0.417	-	-	-	0.507	0.403	**0.523**
DRB5-0101	0.288	0.313	0.322	-	0.330	**0.347**	0.335
H2-IAb	-	-	-	-	-	-	0.724
H2-IAd	-	-	-	-	-	-	0.518
H2-IAs	-	-	-	-	-	-	0.465

“-” indicates performance estimates that are currently unavailable (See text for details).

TABLE IV

AUC values for NetMHCII, MOEA, AND MHCMIR methods evaluated on IEDB benchmark datasets. Results of NetMHCII and MOEA are taken from [10] AND [14], respectively. For each dataset, the rank of each classifier is shown in parentheses.

Dataset	NetMHCII	MOEA	MHCMIR
DRB1-0101	0.716(2)	0.651(3)	0.778(1)
DRB1-0301	0.765(2)	0.778(1)	0.761(3)
DRB1-0401	0.758(2)	0.725(3)	0.760(1)
DRB1-0404	0.785(3)	0.786(2)	0.794(1)
DRB1-0405	0.735(2)	0.756(1)	0.721(3)
DRB1-0701	0.787(1)	0.735(3)	0.754(2)
DRB1-0802	0.756(3)	0.773(1)	0.772(2)
DRB1-0901	0.775(1)	0.712(2)	0.664(3)
DRB1-1101	0.734(2)	0.759(1)	0.734(2)
DRB1-1302	0.818(3)	0.820(2)	0.852(1)
DRB1-1501	0.736(3)	0.743(2)	0.774(1)
DRB4-0101	0.736(3)	0.759(2)	0.801(1)
DRB5-0101	0.664(2)	0.660(3)	0.675(1)
H2-IAb	0.908(2)	0.919(1)	0.894(3)
H2-IAd	0.818(2)	0.855(1)	0.775(3)
H2-IAs	0.898(1)	0.889(2)	0.771(3)
Avg	0.774(2.13)	0.770(1.88)	0.768(1.94)

TABLE V

Comparison of AUC values for different MHC-II peptide prediction methods on IDS-Wang dataset [49]. The results of the best-performing method are highlighted in bold.

Allele	peptides	ARB	MHC2PRED	MHC2PRED	PROPRED	RANKPEP	SMM-align	SVRMHC	SYFPEITHI	MHCMIR
DRB1*0101	3882	0.76	0.71	0.62	0.74	0.70	0.77	0.69	0.71	**0.81**
DRB1*0301	502	0.66	0.53	-	0.65	0.67	**0.69**	-	0.65	0.64
DRB1*0401	512	0.67	0.52	0.630	0.69	0.63	0.68	0.66	0.65	**0.73**
DRB1*0404	449	0.72	0.64	-	**0.79**	0.66	0.75	-	-	0.73
DRB1*0405	457	0.67	0.51	-	**0.75**	0.62	0.69	0.62	-	0.73
DRB1*0701	505	0.69	-	0.63	0.78	0.58	0.78	-	0.68	0.83
DRB1*0802	245	0.74	0.70	-	**0.77**	-	0.75	-	-	0.74
DRB1*0901	412	0.62	0.48	-	-	0.61	**0.66**	-	-	0.62
DRB1*1101	520	0.73	0.60	-	0.80	0.70	0.81	-	0.73	**0.81**
DRB1*1302	289	**0.79**	0.54	-	0.58	0.52	0.69	-	-	0.72
DRB1*1501	520	0.70	0.63	-	0.72	0.62	**0.74**	0.64	0.67	0.73
DRB3*0101	420	0.59	-	-	-	-	**0.68**	-	-	-
DRB4*0101	245	0.74	0.61	-	-	0.65	0.71	-	-	**0.76**
DRB5*0101	520	0.70	0.59	-	**0.79**	0.73	0.75	0.63	-	0.71
IAB	500	**0.80**	0.56	0.51	-	0.74	0.75	-	-	0.69
IED	39	-	-	0.53	-	**0.83**	-	-	-	-
Mean		0.71	0.58	0.58	0.73	0.66	0.73	0.65	0.68	0.73
Min		0.59	0.48	0.51	0.58	0.52	0.66	0.62	0.65	0.62
Max		0.80	0.70	0.63	0.80	0.83	0.81	0.69	0.73	0.83

“-” indicates performance estimates that are currently unavailable (See text for details).

TABLE VI

Identification of MHC-II peptides binding cores in a dataset of 15 HLA-DRB1 restricted peptides. Columns in the table indicate the HLA allele, protein data bank (PDB) identifier, lenth of the peptide, peptide sequence, the binding core as determined from the protein structure, and the binding cores as predicted by NetMHCIIpan and MHCMIR methods. erroneous predicted cores are underlined.

Allele	PDB ID	length	peptide	core	NetMHCIIpan	MHCMIR
DRB1*0101	2FSE	14	AGFKGEQGPKGEPG	FKGEQGPKG	FKGEQGPKG	FKGEQGPKG
DRB1*0101	1KLG	15	GELIGILNAAKVPAD	IGILNAAKV	IGILNAAKV	IGILNAAKV
DRB1*0101	1SJE	16	PEVIPMFSALSEGATP	VIPMFSALS	VIPMFSALS	VIPMFSALS
DRB1*0101	1FYT	13	PKYVKQNTLKLAT	YVKQNTLKL	YVKQNTLKL	YVKQNTLKL
DRB1*0101	1AQD	15	VGSDWRFLRGYHQYA	WRFLRGYHQ	WRFLRGYHQ	FLRGYHQYA
DRB1*0101	1PYW	11	XFVKQNAAALX	FVKQNAAAL	FVKQNAAAL	FVKQNAAAL
DRB1*0101	1T5X	15	AAYSDQATPLLLSPR	YSDQATPLL	YSDQATPLL	YSDQATPLL
DRB1*0301	1A6A	15	PVSKMRMATPLLMQA	MRMATPLLM	MRMATPLLM	VSKMRMATP
DRB1*0401	2SEB	12	AYMRADAAAGGA	MRADAAAGG	YMRADAAAG	YMRADAAAG
DRB1*0401	1J8H	13	PKYVKQNTLKLAT	YVKQNTLKL	YVKQNTLKL	YVKQNTLKL
DRB1*1501	1BX2	15	ENPVVHFFKNIVTPR	VHFFKNIVT	VHFFKNIVT	VHFFKNIVT
DRB1*1501	1YMM	23	ENPVVHFFKNIVTPRGGSGGGGG	VHFFKNIVT	VHFFKNIVT	VHFFKNIVT
DRB5*0101	1H15	14	GGVYHFVKKHVHES	YHFVKKHVH	YHFVKKHVH	FVKKHVHES
DRB5*0101	1FV1	20	NPVVHFFKNIVTPRTPPPSQ	FKNIVTPRT	FKNIVTPRT	FFKNIVTPR
DRB5*0101	1ZGL	15	VHFFKNIVTPRTPGG	FKNIVTPRT	FKNIVTPRT	FFKNIVTPR

TABLE VII

Comparison of AUC values of MILESreg predictors evaluated using 5-fold cross-validation on IEDB dataset using bags of 9, 10, 11, and 12 -mers, respectively. The rank of each classifier is shown in parentheses.

Allele	9-mer	10-mer	11-mer	12-mer
DRB1*0101	0.780(1.0)	0.776(4.0)	0.778(2.5)	0.778(2.5)
DRB1*0301	0.772(1.0)	0.763(2.0)	0.758(3.0)	0.741(4.0)
DRB1*0401	0.774(1.0)	0.765(2.0)	0.754(3.0)	0.731(4.0)
DRB1*0404	0.806(1.0)	0.792(2.0)	0.786(3.0)	0.749(4.0)
DRB1*0405	0.715(1.0)	0.710(2.0)	0.709(3.0)	0.652(4.0)
DRB1*0701	0.744(4.0)	0.761(2.0)	0.766(1.0)	0.748(3.0)
DRB1*0802	0.779(2.0)	0.787(1.0)	0.774(4.0)	0.775(3.0)
DRB1*0901	0.713(3.0)	0.756(1.0)	0.714(2.0)	0.687(4.0)
DRB1*1101	0.758(1.5)	0.753(3.5)	0.753(3.5)	0.758(1.5)
DRB1*1302	0.850(1.0)	0.827(2.0)	0.806(4.0)	0.807(3.0)
DRB1*1501	0.781(1.0)	0.780(2.0)	0.761(4.0)	0.763(3.0)
DRB4*0101	0.821(1.0)	0.798(2.0)	0.778(4.0)	0.786(3.0)
DRB5*0101	0.708(3.0)	0.727(1.0)	0.705(4.0)	0.722(2.0)
H2-IAb	0.924(1.0)	0.900(2.0)	0.846(3.0)	0.817(4.0)
H2-IAd	0.791(3.0)	0.797(1.5)	0.797(1.5)	0.775(4.0)
H2-IAs	0.843(2.0)	0.850(1.0)	0.839(3.0)	0.819(4.0)
Avg. ranks	0.785(1.72)	0.784(1.94)	0.77(3.03)	0.757(3.31)

Algorithm 1

Training MILESreg

```
**Input** : $B = \{\langle B_1, y_1\rangle, \ldots \langle B_m, y_m\rangle\}$ set of training bags
Let $C = \{x^1, \ldots, x^n\}$ be set of all instances extracted from $B$
**for all** $i$ such that $\langle B_i, y_i\rangle \in B$ **do**
    Let $I_i$ be a new instance of $n$ attributes
    **for all** $k$ such that instance $x^k \in C$ **do**
        Set $k^{th}$ attribute in $I_i$ to $min_j \, dist(x_{ij}, x^k)$
    **end for**
**end for**
Build an SVR model using meta instances $I$

**Function:** dist
**Parameters** : $s1$ and $s2$ two 9-mer subsequences
**for** $i = 1$ to 9 **do**
        $d+ = BLOSUM62(s1[i], s2[i])$
**end for**
**if** $(d \leq 0)$ **then**
        **return 1**
**else**
        **return** $\frac{1}{d}$
**end if**
```