PREDICTION OF PEPTIDES BINDING TO MHC CLASS I ALLELES BY PARTIAL PERIODIC PATTERN MINING

Cem Meydan[1], Hasan Otu, Uğur Sezerman[1]

[1] Biological Sciences and Bioengineering Dept., Sabanci University
Tuzla, 34956, Istanbul, Turkey
phone: + (90) 216 483 9513, fax: + (90) 216 483 9550,
email: cemmeydan@su.sabanciuniv.edu, hotu@bidmc.harvard.edu, ugur@sabanciuniv.edu

ABSTRACT

MHC (Major Histocompatibility Complex) is a key player in the immune response of an organism. It is important to be able to predict which antigenic peptides will bind to a specific MHC allele and which will not, creating possibilities for controlling immune response and for the applications of immunotherapy. However a problem encountered in the computational binding prediction methods for MHC class I is the presence of bulges and loops in the peptides, changing the total length. Most machine learning methods in use today require the sequences to be of same length to successfully mine the binding motifs. We propose the use of time-based data mining methods in motif mining to be able to mine motifs position-independently. Also, the information for both binding and non-binding peptides are used on the contrary to the other methods which only rely on binding peptides. The prediction results are between 70-80% for the tested alleles.

1. INTRODUCTION

MHC (Major Histocompatibility Complex) is a large gene family with an important part of the immune system, autoimmunity and reproduction. MHC molecules take role in destruction of pathogens and diseased cells by showing self and non-self antigen peptides on their surface and coordinating the T-cells which identify these peptides. The T-Cells recognize the infected cell upon binding to the antigenic peptide-MHC complex and trigger the immune response to foreign bodies by a cascade of events. Since they have a key role in immune response, MHCs are critical in many diseases, and they can be used for controlling specific processes by creating peptides to bind to specific MHC alleles. This binding affinity to specific peptides may be exploited for creating peptide vaccines, suppressing specific alleles in organ transplants, and many other possible areas in immunotherapy.

Although the peptide lengths for MHC class I proteins is usually only 8-10 aminoacids long, they show great variance within their composition. The peptide binding groove in the MHC molecules binds peptides with high promiscuity; it is estimated that each HLA (human leukocyte antigen system) class I protein can bind between 1000 and 10,000 peptides. Thus it is difficult to find specific motifs for experimental studies, and the large number of possible structures makes it infeasible to find them by experiments alone. Computational determination of binding specificity of a given peptide to specific alleles is an important problem in bioinformatics. Although many methods have been proposed, still the accuracy is not near what can be expected for such short motifs. The most state of art prediction servers can predict alleles with 75-95% accuracy for easy classes to 50-65% for hard classes [1, 2]. This area is still open to research.

These methods usually depend on 2 to 3 specific anchors that it checks for, and if these critical aminoacids are found, it scores the remaining aminoacids. The difficult classes of peptides show bulges and loops in their structure, changing the length of the peptide from the optimal length of 9. These bulges shift the binding points of the peptide, and thus make the position specific motif finding and scoring methods, which show acceptable accuracy in easy classes, unusable. We aim to include a novel method for extracting the motifs that are varied due to the bulges that can be used on difficult sets. Our proposed method is to apply sequence mining domain of data mining for ordered episodes, which was previously untried. These temporal mining algorithms are usually used in intrusion detection and other future prediction methods, which try to capture the patterns which occur in an order but not necessarily consecutively. These partial periodic pattern mining methods could help tackle the problem of shifted binding position in position specific methods.

2. BACKGROUND

Computational methods for prediction of the binding affinity of a peptide to an MHC allele are based on three main artificial learning systems, namely statistical, structural and neural methods [3-5].

Statistical methods extract the statistical properties of different class values and use classification algorithms on these properties. Structural methods mine the common structural motifs and properties within the same class and/or the differences in different classes. Neural methods such as artificial neural networks are based on a network structure in which the connected units (neurons) are activated according to the input signals propagating through the network [6]. Artificial neural networks can capture very complex relationships, can tolerate outliers and erroneous data and deal with non-linear relationships [3, 6, 7]. Due to their adaptive nature, they can fit to very complex relationships. However, they require

more data than other methods to accurately learn the large number of parameters the nodes have.

Sequence similarity of a new peptide to the known binders/non-binders using alignment methods [8], is one of the simplest, but least accurate methods. In the binding groove of the MHC alleles, specific positions affect the binding affinity more than the others, thus the independent scoring of matches and mismatches in different positions, as well as the lack of substantial penalty for not having specific aminoacids in key positions makes it non-suitable for MHC prediction. To overcome these, methods using binding motifs are developed [9]. These motifs describe aminoacids occurring in a specific position that commonly bind to a specific MHC allele, to create a weighted scoring systems in which specific residues at key positions are given favorable scores to bind, as well as negative scores for penalizing residues disrupting the binding process [10-14]. Although the exact accuracy changes from method to method, binding motifs can be very accurate depending on the variety of the binders to a specific MHC allele.

Another sequence similarity method is the quantitative matrices which use position weight matrices (PWM). Each column is a specific position of the finite length motifs and each row is a specific residue. The matrix is filled from the probabilities of seeing a specific residue in a specific position in the training data, and the binding probability of a new peptide is found by scoring its residues [15, 16]. Although it can find weaker patterns than the binding motif methods, overtraining is common. Also, the independent contribution of each position to the score can cause some amount of misclassification [17].

Similar to ANN in principle, Hidden Markov models (HMM) are finite state models which carry the statistical information about the training peptides. They carry the probabilistic state transition information for shifting between different probability distributions of residues for specific states [18]. HMMs can find strong motifs easily, and are better suited to motifs of arbitrary length, in contrary to other methods requiring a fixed-length sequence.

Apart from the sequential information, structural information can also be used. One example to the structural methods is the molecular modelling. Molecular modelling uses the 3-D structure of the MHC molecules to predict the binding energy of a peptide by the protein-peptide interactions. Threading [19], ab initio methods and molecular dynamics can give very accurate results, and also help in the understanding of the given processes. However, they require much more computational power, and the number of peptides with known 3-D structure is relatively low, although computational prediction methods can help even in the lack of any training sequence. Also, these methods can be extended to use structural alignment and structural motif mining algorithms on the 3-D data to find structural binding motifs.

Thus, various methods are employed for MHC binding peptide prediction, some combinations of others [20]. However, for many methods a length constraint of 9 is forced due to the majority of MHC Class I peptides being 9 aminoacids long, although 8-10-11 long binding peptides can also be present due to the presence of only few allele bound aminoacids. Other aminoacids may form a fold, giving rise to peptides of length different than 9, shifting the positions of the aminoacids in the anchor locations. This difference causes the position-specific scoring matrices or other position-dependent methods to fail, for both learning from and the prediction/generation of shorter and longer peptides. Newer methods use results of the sampling of random insertions for elongation and deletions for shortening, meant for fitting the peptide into the 9-length window, thus the 9 limitation is still present in the core. ANN, quantitative matrices, most binding motif miners and methods relying on sequence information requires the peptides to be in the same length, with appropriate peptides aligned to be in the same location. However since the actual peptide data can change between 8-11, and sometimes even reaching more than 20 aminoacids long, these methods require pre-processing and complex alignment of the data to get reasonable results. However, this preprocessing may not be always feasible or give good results on the training set. For this reason, we propose a method which does not require the peptides to be of same length and the anchor positions to be specific, using partial periodic pattern mining. This allows the mining of motifs that have loops and insertions between the anchor positions. Another novel part is the use of both binding and non-binding motif information concurrently.

3. METHODS

3.1. Motif Mining

The motif mining algorithm is based on the apriori algorithm that is used in frequent itemset discovery. Apriori algorithm uses the principle that all subsets of a frequent itemset must also be frequent. Accordingly, it has a bottom-up approach where the shorter frequent itemsets are extended to create longer candidates, which are then filtered by frequency of occurrence [21-23]. This iterative extension process continues until no frequent itemsets of a length can be found. The ratio of the sequences containing the rule to all of the sequences is called the support of the rule, and the ratio of the sequences containing the extended rule to the parent rule is called the confidence.

Our motif mining method is similar to temporal event mining in time-related databases. In general, the partial periodic pattern mining algorithms for time series data will try to find frequently co-occurring events, or causality relationships between them. In the domain of protein motifs, the aminoacids become the “events”, and the causality/future prediction aspects become the motifs that are sought [24].

In the approach, each sequence is taken as a separate time series, with many parallel events occurring at the same time, although each event is related with only the sequence it is found on. In these time series, if an event happens frequently after another event occurs, within a given time window, it is considered an episode of events, a motif. To exploit the apriori principle for performance, the motifs are started from length 1. A longer motif including a specific aminoacid will have support less than or equal to the support of that aminoacid, and if an aminoacid is not frequent, any motif that includes that aminoacid will, therefore will not be frequent.

Due to this, only some combinations of rules are tried to increase performance.

First the frequent itemsets of size 1, F(1), are found, which in our case are the aminoacids. The first step is straightforward, only the aminoacid counts at different positions within the sequences are counted, and if their frequency (support of the rule) is below the given threshold, they are filtered out. Then the candidate set of size 2, C(2) is created from the aminoacids by F(1) → F(1), which is filtered according to the minimum support and confidence values given, creating F(2). Thus, iteratively F(n) is created from filtering of C(n)=F(n-1)→F(1). Notice that we always add F(1) in the extension step, this is due to the fact F(n) for n>1 can be created from F(1) in n steps. Adding F(2), F(3) along F(1) in extension step may seem as to increase the performance, however since the peptides that bind to MHC-I are very short, our motifs are predominantly at most 3 or 4 aminoacids long. Thus, they will create longer rules which are not frequent, increasing the number of candidate set and affecting the performance negatively.

The actual apriori algorithm is more complex due to the intractable nature of great number of transactions, however in our case there are only 20 possible items (aminoacids) with about few thousands of transactions (sequences) at most, thus are not necessary. Also it is modified to allow for ordered sequences unlike the shopping basket subset approach. For making the sequence mining robust, a specific window is defined as being between at least (minimum space) away and at most (maximum space) away. If the aminoacids co-occur within this window by a specific order, at least (minimum support x sequence count) times, then it is considered frequent.

In the motif mining context, the frequent rules are not association rules as in a shopping basket analysis, they have a time value which is used for relations such as "before"/"after" ("simultaneously" is not used in protein motifs since at each time point, that is a specific position in the sequence, only one aminoacid can occur). Then the episodes become, "if A occurs in a given position, B will likely to occur within n to m positions after A with probability of p and confidence of c". There are two parameters, the slack length (s), which is the length after an event within which we do not look for a rule, and the window size (w), in which the consequent event may occur. Thus, n=s+1 and m=s+w-1 in the above definition, and the rule is given as A→B (p, c) for parameters (s, w). The rule may also consist of 3 or longer events, such as A→B→C.

While experimenting, we used window size of 3 and slack length of 0 to 8, which produced different rulesets. For s=0, the rules that consist of consecutive/nearby aminoacids are mined whereas for larger values of s, the motifs consisting of aminoacids at separate ends of the peptide are found. Since the anchor positions of MHC motifs may be different, different slack lengths are needed to mine them all.

3.2. Prediction

Once the rules are mined, these rules are used in the prediction and scoring process. Before prediction, rules from both the binding and non-binding sequences are mined separately. During classification of an unknown peptide, the peptide is scored independently by both of the binding and non-binding rules. The simplest classification method is the direct comparison of the scores for binding/non-binding. To calculate the scores, the support values of the rules that occur in the given peptide are summed for both classes. However, the binding and non-binding datasets are usually not balanced due to the very low count of non-binding peptides. For example, the allele HLA-A0201, one of the most studied alleles, has 1390 binding sequences but only 60 non-binding sequences. Most alleles are less than even 60, with some having only 0-5 non-binding peptides. Due to these low non-binding counts, many rules exceed the minimum support threshold. While mining rules from HLA-A0201 with a minimum support of 0.1 (value used in the experiments), a rule need to be present in 139 positive peptides to be included, whereas it only needs to occur 6 times in negative class. Due to this imbalance, the rule count for negative class is substantially higher. If we are to sum the support values, the addition of non-binding rules usually results in higher nonbinding values for both classes. To overcome this problem, different minimum support values can be used, or the highest ranking *N* rules may be taken. However, these cause the loss of information, thus we tried balancing the sums of all of the rules. Sum of both classes are equalized to some value. Hence, for two rules with the same support value, one that is found in the dataset with the lower count of rules has a higher score, considering that rule is much important for that class separation than the other. However, in the normalization of the sum of the rules, the sum of the both classes needn't be equal, for a training dataset an optimal multiplier for both binding and nonbinding may be found that separates the scores with the greatest threshold. We added an optimization step for the weights for positive and negative classes and also the best cut-off value to use as a threshold for class separation.

4. RESULTS

4.1. Data Set

The dataset used is MHCBN from Raghava et al. [25]. The total database consists of 25860 peptides, 20717 binders and 4022 non-binders. The alleles HLA-A0201, HLA-A2 and HLA-DR2 are used in testing. The binding affinity values of high/medium/low are combined to create the binder dataset and the rest are taken as non-binder for a binary value. The actual affinity values are not used in the mining/scoring process.

4.1. Experiments

As we said, the binding and non-binding datasets are not balanced, and according to Sales et al., unbalanced datasets reduce the accuracy dramatically. However, resampling the non-binding peptides or undersampling the binding peptides does not increase the accuracy and sometimes decrease it as well [26]. To overcome this problem, we used the binding peptides to generate non-binding samples. While the patterns for non-binding can be mined by looking at what occurs in

DataSet		Accuracy			Sensitivity			Specificity			Precision		
		Avg	Max	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min
HLA-A0201 (1390 Pos, 60 Neg)	**Train**	0.806	0.838	0.757	0.808	0.843	0.754	0.756	0.854	0.646	0.987	0.992	0.984
	Test	0.794	0.852	0.762	0.802	0.860	0.773	0.620	0.917	0.333	0.980	0.995	0.964
HLA-A0201 + 100 Synthetic Neg	**Train**	0.807	0.876	0.712	0.808	0.890	0.708	0.771	0.854	0.708	0.988	0.992	0.978
	Test	0.795	0.883	0.728	0.804	0.914	0.720	0.607	0.917	0.417	0.979	0.995	0.962
HLA-A2 (682 Pos, 222 Neg)	**Train**	0.720	0.751	0.684	0.714	0.772	0.655	0.739	0.853	0.684	0.897	0.919	0.883
	Test	0.747	0.808	0.676	0.809	0.891	0.715	0.556	0.733	0.378	0.849	0.888	0.803
HLA-DR2 (407 Pos, 174 Neg)	**Train**	0.681	0.716	0.601	0.662	0.754	0.517	0.726	0.871	0.597	0.852	0.904	0.816
	Test	0.620	0.692	0.513	0.549	0.646	0.354	0.784	0.943	0.629	0.859	0.954	0.797

Table 1: The results for the prediction of 3 MHC class I alleles, with 80% training, 20% testing set separation repeated for 25 times.

non-binding sequences, they can also be mined by looking at what does not happen in the binding peptides. Since the binding peptide count is high, the distribution of the aminoacids on a specific position was found and a new sequence was generated with aminoacids inversely proportional to the ones found in the binding sequence. Thus, for example, if none or very few of the peptides binding allele HLA-A0201 have { D, E, R, K } in position 3, then it is likely that these aminoacids are negatively affecting the binding affinity of the peptide [11]. Since it is possible that the non-peptides are not varied enough to capture this pattern, newly generated non-binder sequences can help in this process. However care must be taken to not suppress the actual non-binding sequences since there is no guarantee that the generated sequences actually have patterns that help in the classification. A higher support threshold may reduce the possible random patterns. In our experiments, the synthetic dataset increased the accuracy slightly in some cases, but decreased it more significantly for some datasets, thus their effect should be controlled.

For testing, 3 alleles, HLA-A0201, HLA-A2 and HLA-DR2 are used based on the availability of sufficient binder/non-binder data and importance in the literature. For HLA-A0201, the ratio of positive to negative class was about 23 to 1, to balance this ratio to more acceptable levels without under-representing the actual non-binder data, an additional of 100 synthetic non-binder peptides were created to compare the effects with and without these synthetic peptides. Each allele is tested by dividing the data into 80% training 20% testing sets randomly, a total of 25 times for an allele. The average, maximum and minimum results for the 4 datasets, with both training and testing set accuracies can be found in Table 1.

It can be seen that the predictions have acceptable accuracy values of 70 to 80%. Note that the specificity is about 5-10% lower than accuracy (high false positive rate) in HLA-A0201. On the contrary, for HLA-DR2 the specificity is ~5-10% higher than accuracy. For HLA-A0201, the false positive rate is the result of low non-binding count. If we look at the peptides that are classified as binding, when in fact non-binding, they carry very strong binding patterns, such as the L→{L-I-V} pattern in anchor positions of HLA-A0201. These aminoacids in the 2-9 positions are accepted by the literature as good binders. The peptides that are classified as false positives carry these patterns and other strong patterns. It is obvious that they carry another part that suppresses the affinity of the binding motif to the MHC allele. While the method marks some peptides with good positive scores as non-binding due to the presence of a non-binding signal, it cannot catch them all, possibly due to the lower count of the negative dataset. However an important point to consider is that the non-binding accuracies given do not reflect the whole domain of the non-binders, since the dataset has an experimental bias. The sequences tested and marked as non-binding are either poly-alanine sequences or known binding sequences, on which mutations are carried out repeatedly to find the binding position and rules. If a binding peptide becomes non-binding (or a poly-A peptide becomes binding) after introducing a mutation into a specific site, that aminoacid in that position will be deemed important. But this data, in training can cause artificial patterns to be found. The prediction method will accurately find non-binders that do not carry the binding patterns as well, which are under-represented in this dataset, showing a lower negative prediction accuracy than the actual value.

Poly-A peptides can also affect the accuracy. Since sequences consisting of 6-7 alanine residues are very common, these can cause the artificial pattern of (A→A), (A→A→A) and similar to be found. The problems caused by this are relatively less important, because both the binders and non-binders carry this pattern, thus making it inconclusive for classification. However if these poly-A peptides are under- or over-represented in one dataset, it will start to affect the accuracy. Filtering these sequences, as some other methods do, may also help.

5. CONCLUSION

We developed a method that uses sequential pattern mining schemes for finding the most probable binding motif, with position- independent information that can be applied to the peptides of arbitrary length to accommodate for the sequences with insertions and loops between the anchor positions. The frequent partial periodic rules that can explain most of the peptides are mined from the training set using different windows for position-independent episodes. For the same allele, the non-binding peptide information is also used for mining motifs for non-binding, since the mined episodes may contain arbitrary episodes that are not related to the binding affinity. Also, some additional peptides in the non-

binding aminoacid positions may cause a previously binding peptide to become non-binding. Thus, we mined frequent rules for both binding and non-binding peptides, and use the exclusive set of the two for scoring the peptides. The peptides are scored according to the support and confidence of the frequent episodes they contain. From this study, position independent motifs mined with representing the aminoacid sequence as time series data proved to be usable for prediction of the binding peptides to MHC class I proteins. Although the accuracy of the algorithm is not state-of-the-art, it is acceptable and near the methods in use today. The pattern mining method can be improved upon to include some position dependency as anchor points or windows, and by the addition of rule merging/splitting for better class separation.

The area most open to improvement is the scoring and prediction step; while the rules mined are quite representative of the classes, the current scoring system is very basic and is prone to make errors. The prediction process can be changed to include better scoring system, rule cascading and weighting to differentiate between strict and more relaxed rules, and statistical methods to select rules with which the peptide is scored, to improve the statistical significance of the given prediction. The rules may also be created such to estimate the binding affinity of the peptide, instead of a binary binding/non-binding decision. This will both improve the learning step, which will give more weight to the peptides with high affinity, and the classification step, giving more information about the stability of bonds in the MHC.

An area worth exploring are the generation of the synthetic non-binder peptides which can increase the real-world screening accuracy by creating more representative sequences. Novel methods as well as known methods in the literature may be explored and optimized for this motif mining algorithm.

REFERENCES

[1] Lin HH, Ray S, Tongchusak S, Reinherz EL, Brusic V. Evaluation of MHC class I peptide binding prediction servers: applications for vaccine research. BMC Immunol. 2008;9:8.

[2] Peters B, Bui HH, Frankild S, Nielson M, Lundegaard C, Kostem E, et al. A community resource benchmarking predictions of peptide binding to MHC-I molecules. PLoS Comput Biol. 2006 Jun 9;2(6):e65.

[3] Schalkoff RJ. Pattern recognition : statistical, structural, and neural approaches. New York: J. Wiley 1992.

[4] Brusic V, Bajic VB, Petrovsky N. Computational methods for prediction of T-cell epitopes--a framework for modelling, testing, and applications. Methods. 2004 Dec;34(4):436-43.

[5] Firebaugh MW. Artificial intelligence : a knowledge-based approach. Boston: Boyd & Fraser 1988.

[6] Zurada JM. Introduction to artificial neural systems. St. Paul: West 1992.

[7] Weiss SM, Kulikowski CA. Computer systems that learn : classification and prediction methods from statistics, neural nets, machine learning, and expert systems. San Mateo, Calif.: M. Kaufmann Publishers 1991.

[8] Celis E, Larson J, Otvos L, Jr., Wunner WH. Identification of a rabies virus T cell epitope on the basis of its similarity with a hepatitis B surface antigen peptide presented to T cells by the same MHC molecule (HLA-DPw4). J Immunol. 1990 Jul 1;145(1):305-10.

[9] Falk K, Rotzschke O, Stevanovic S, Jung G, Rammensee HG. Allele-specific motifs revealed by sequencing of self-peptides eluted from MHC molecules. Nature. 1991 May 23;351(6324):290-6.

[10] Disis ML, Smith JW, Murphy AE, Chen W, Cheever MA. In vitro generation of human cytolytic T-cells specific for peptides derived from the HER-2/neu protooncogene protein. Cancer Res. 1994 Feb 15;54(4):1071-6.

[11] Houbiers JG, Nijman HW, van der Burg SH, Drijfhout JW, Kenemans P, van de Velde CJ, et al. In vitro induction of human cytotoxic T lymphocyte responses against peptides of mutant and wild-type p53. Eur J Immunol. 1993 Sep;23(9):2072-7.

[12] Meister GE, Roberts CG, Berzofsky JA, De Groot AS. Two novel T cell epitope prediction algorithms based on MHC-binding motifs; comparison of predicted and published epitopes from Mycobacterium tuberculosis and HIV protein sequences. Vaccine. 1995 Apr;13(6):581-91.

[13] Ruppert J, Kubo RT, Sidney J, Grey HM, Sette A. Class I MHC-peptide interaction: structural and functional aspects. Behring Inst Mitt. 1994 Jul(94):48-60.

[14] Stauss HJ, Davies H, Sadovnikova E, Chain B, Horowitz N, Sinclair C. Induction of cytotoxic T lymphocytes with peptides in vitro: identification of candidate T-cell epitopes in human papilloma virus. Proc Natl Acad Sci U S A. 1992 Sep 1;89(17):7871-5.

[15] Parker KC, Bednarek MA, Coligan JE. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. J Immunol. 1994 Jan 1;152(1):163-75.

[16] Schafer JR, Jesdale BM, George JA, Kouttab NM, De Groot AS. Prediction of well-conserved HIV-1 ligands using a matrix-based algorithm, EpiMatrix. Vaccine. 1998 Nov;16(19):1880-4.

[17] Peters B, Tong W, Sidney J, Sette A, Weng Z. Examining the independent binding assumption for binding of peptide epitopes to MHC-I molecules. Bioinformatics. 2003 Sep 22;19(14):1765-72.

[18] Baldi P, Brunak S. Bioinformatics : the machine learning approach. 2nd ed. Cambridge, Mass.: MIT Press 2001.

[19] Jojic N, Reyes-Gomez M, Heckerman D, Kadie C, Schueler-Furman O. Learning MHC I--peptide binding. Bioinformatics. 2006 Jul 15;22(14):e227-35.

[20] Trost B, Bickis M, Kusalik A. Strength in numbers: achieving greater accuracy in MHC-I binding prediction by combining the results from multiple prediction tools. Immunome Res. 2007;3:5.

[21] Srikant R, Agrawal R. Mining generalized association rules. Future Generation Computer Systems. 1997 Nov;13(2-3):161-80.

[22] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules in Large Databases. *Proceedings of the 20th International Conference on Very Large Data Bases*: Morgan Kaufmann Publishers Inc. 1994.
[23] Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*. Washington, D.C., United States: ACM 1993.
[24] Mannila H, Toivonen H, Verkamo AI. Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery. 1997 Nov;1(3):259-89.
[25] Bhasin M, Singh H, Raghava GP. MHCBN: a comprehensive database of MHC binding and non-binding peptides. Bioinformatics. 2003 Mar 22;19(5):665-6.
[26] Sales AP, Tomaras GD, Kepler TB. Improving peptide-MHC class I binding prediction for unbalanced datasets. BMC Bioinformatics. 2008;9:385.