

Building a Meta-predictor for MHC Class II Binding Peptides

Lei Huang[¶], Oleksiy Karpenko[¶], Naveen Murugan[¶], and Yang Dai*

Lei Huang · Department of Bioengineering, The University of Illinois at Chicago,
Chicago, IL

Oleksiy Karpenko · Department of Bioengineering, The University of Illinois at Chicago,
Chicago, IL

Naveen Murugan · Department of Bioengineering, The University of Illinois at Chicago,
Chicago, IL

Yang Dai · Department of Bioengineering, The University of Illinois at Chicago, Chicago,
IL

*Corresponding author:

Yang Dai, PhD
Dept. of Bioengineering (M/C063)
University of Illinois at Chicago
851 South Morgan Street
Chicago, IL 60607
Phone: (312) 413-1487
Fax: (312)413-2018
Email: yangdai@uic.edu

[¶]These authors contributed equally.

Abstract

Prediction of class II MHC-peptide binding is a challenging task due to variable length of binding peptides. Different computational methods have been developed; however, each has its own strength and weakness. In order to provide reliable prediction, it is important to design a system that enables the integration of outcomes from various predictors. In this chapter, the procedure of building such a meta-predictor based on Naïve Bayesian approach is introduced. The system is designed in such a way that results obtained from any number of individual predictors can be easily incorporated. This meta-predictor is expected to give users more confidence in the prediction.

Key Words: MHC class II binding, epitope prediction, meta-predictor, Naïve Bayesian classifier

1. Introduction

T cell-mediated immune responses are initiated by the activation of effector T cells. The activation process requires the recognition of the complex formed between an antigen peptide and a major histocompatibility complex (MHC) protein by the T cell receptor. The identification of peptides that bind to MHC molecules plays a crucial role in understanding the mechanisms of both humoral and adaptive immunity as well as developing epitope-based vaccines. Experiments for measuring the binding affinities of peptides to MHC molecules are time consuming and expensive. It is a prohibitive task to identify potential binding peptides from the host and pathogen proteins on a genome-wise

scale. Therefore, considerable efforts have been made on the development of computational tools for the identification of MHC-binding peptides (1, 2).

Two major types of MHC molecules are involved in the peptide binding process. MHC class I molecules present endogenous antigens (e.g. viral peptides or tumor antigens synthesized within the cytoplasm of a cell) to CD8+ cytotoxic T cells. MHC class II molecules, on the other hand, present exogenously derived proteins (e.g. bacterial proteins or viral capsid proteins) through antigen presenting cells (APC) to CD4+ helper T cells (3). Generally, antigen peptides that bind to both MHC class I and class II molecules are approximately nine amino acid residues long. However, the peptide-binding groove of a MHC class II molecule is open at both ends, which makes it capable of accommodating longer peptides of 10-30 residues (4-6).

The length variability complicates the prediction of peptide-MHC class II binding. However, analyses of the binding motif and the structure of peptide-MHC class II complexes have suggested that a core of 9 residues within a peptide is essential for peptide-MHC binding. Computational methods for the prediction include simple binding motifs (7, 8), quantitative matrices (9), hidden Markov models (10), artificial neural networks (11, 12) and support vector machines (13). Some of these methods require a preprocessing step to align binding sequences with various lengths for the identification of subsequences of the binding cores. Since each method has its own strength and weakness, it is hard for an immunologist to select a single method from the pool of existing predictors. Therefore, a system that produces reliable prediction through the integration of outcomes from major prediction methods is in clear need.

In this chapter, the steps for building such a system based on the Naïve Bayesian (14) approach are presented. The Bayesian framework has the flexibility to incorporate any predictor that makes prediction from a computed score correlated with the binding affinity of MHC class II peptides. Here, in order to illustrate the steps of the Bayesian framework, three individual predictors, i.e., ProPred, the Gibbs sampler, and the LP model are selected.

ProPred, designed by Singh and Raghava (15), applied the quantitative matrices from 51 HLA-DR alleles for the prediction of MHC class II binding peptides. These matrices were generated from a pocket profile database described by Sturniolo et al. (9) and covered the majority of human HLA-DR specificity.

Nielsen, et al. (16) proposed an advanced motif sampler method based on the Gibbs sampling technique, which efficiently samples the possible alignment space of binder sequences. For each alignment a log-odds weight matrix was calculated for the identified binding core subsequences. This matrix serves as the position-specific scoring matrix for the computation of a score for a nonamer.

Motivated by a text mining model designed for building a classifier from labeled and unlabeled examples, Murugan and Dai (17) developed an iterative supervised learning model for the prediction of MHC class II binding peptides. The iterative learning model, based on linear programming (LP), enables the use of non-binder information for the detection of the binding cores from a set of putative binding cores and for the construction of the predictor simultaneously. The outcome of this predictor is a position specific weight matrix that can score amino acids at each position of a nanomer.

2. Materials

1. A dataset that includes binding and nonbinding peptides for a specific MHC class II allele. The recommended size of binders is above 100. Any in-house peptide set can be used. If the number of peptides is not sufficient, peptides from databases such as AntiJen (18) and MHCBN (19) can be added for training. For some alleles, the number of nonbinders may be extremely small. In this case, the random sequences can be added (*see Note 1*).
2. Predictors that can score the binding ability for each individual peptide (*see Note 2*).

3. Methods

The Bayesian predictor is trained based on the prediction outcome obtained from each individual predictor for a set of training peptides. The system is flexible to incorporate results from any number of predictors. Suppose that the number of predictors is m . In general, the requirement for each predictor is the generation of a score for a given peptide sequence. This score of a peptide is designated as the highest value among all scores that are assigned to the overlapping 9-mers of the peptide by a predictor. A peptide is predicted as a binder (resp. nonbinder) if this score is above (resp. below) a prescribed threshold value. The steps for building a Bayesian predictor are given as follows.

1. Prepare a training dataset. Any peptide sequence with length less than nine residues or with undetermined residues in certain positions should be discarded.
2. Reduce the redundancy in the dataset. This step is to prevent overestimation of the performance of a predictor. After the reduction there should be no two peptide

sequences in the set with sequence identity >90% over an alignment of length at least nine residues.

3. Obtain a predictive score for each peptide in the training set (including binding and nonbinding sequences) from each individual predictor. These scores form the input set from which a Bayesian predictor can be built.
4. Determine a set of threshold values that produce distinct pairs of sensitivity and specificity (*see Note 3*). This procedure should be performed for each predictor on the training set. Upon the completion of this step, a set of threshold values for predictor j is obtained, say $\delta^j = (\delta_1^j, \dots, \delta_{t_j}^j)$, $j=1, \dots, m$, where t_j is the number of possible threshold values with the above property for predictor j .
5. Determine the best combination $\delta^* = (\delta^{*1}, \dots, \delta^{*m})$ of threshold values, where each δ^{*j} ($j=1, \dots, m$) is the selected threshold value for predictor j . This combination can be determined by finding the highest average AROC (*see Note 3*) value for the Bayesian predictor with a k -fold cross-validation procedure described as follows.
 - a. For each combination of threshold values $(\delta_{i_1}^1, \dots, \delta_{i_m}^m)$, set up a prediction outcome table for the $(k-1)$ -folds of the training peptides (*see Note 4*), where $\delta_{i_j}^j$ is the i_j th threshold value for predictor j , $j=1, \dots, m$ and $i_j=1, \dots, t_j$. This table is of size $n \times m$, where n is the number of peptides in the training folds. The outcome obtained from predictor j for a peptide is denoted by a binary number f_j : $f_j=1$ if the peptide is predicted as binder, $f_j=0$ otherwise. Accordingly, the

prediction outcome obtained from the m predictors for each peptide will be coded by a binary string $f_1f_2\dots f_m$.

- b. Build the probability table for the Bayesian predictor from the $n \times m$ table described above. Let y_i denote the label of each peptide: $y_i=1$ if it is a binder, $y_i=-1$ if it is not a binder. The probabilities for each value f_j of the m features for the binder class and the nonbinder class are computed as follows.

$$p(f_j = 1 \mid \text{binder class}) = \frac{\sum_{i:y_i=1} I(f_{ij} = 1)}{\text{total number of binders}}, \quad j = 1, \dots, m,$$

$$p(f_j = 0 \mid \text{binder class}) = \frac{\sum_{i:y_i=1} I(f_{ij} = 0)}{\text{total number of binders}}, \quad j = 1, \dots, m,$$

$$p(f_j = 1 \mid \text{nonbinder class}) = \frac{\sum_{i:y_i=-1} I(f_{ij} = 1)}{\text{total number of nonbinders}}, \quad j = 1, \dots, m, \text{ and}$$

$$p(f_j = 0 \mid \text{nonbinder class}) = \frac{\sum_{i:y_i=-1} I(f_{ij} = 0)}{\text{total number of nonbinders}}, \quad j = 1, \dots, m,$$

where $I(\cdot) = 1$ if the condition in the parenthesis is true; $I(\cdot) = 0$ otherwise.

Note that (i) the total numbers of binders and nonbinders are respectively those in the $(k-1)$ training folds; (ii) the index i in the numerator of each formula runs through all peptides in the $(k-1)$ training folds; and (iii) f_{ij} is the prediction by predictor j for the 9-mer with the highest score from peptide i .

- c. For each overlapping 9-mer s_i of a peptide x from the testing fold, compute the ratio of probabilities

$$R_i = \frac{p(f = 1 \mid s_i)}{p(f = 0 \mid s_i)} = \frac{\prod_{j=1}^m p(f_{ij} \mid \text{binder class})}{\prod_{j=1}^m p(f_{ij} \mid \text{nonbinder class})},$$

and select the highest one as the ratio R_x of the peptide x . Here f_{ij} is the prediction outcome obtained from predictor j for 9-mer s_i . This formula is a straightforward application of the Bayesian rule, without the inclusion of the ratio of prior probabilities $p(\text{binder})$ and $p(\text{nonbinder})$. The influence of prior probabilities on prediction will be implicitly considered through threshold of ratio R_i . With a prescribed threshold δ_B for the Bayesian predictor, the peptide is predicted as a binder if R_x is greater than δ_B , otherwise a nonbinder. Varying the threshold values for δ_B , the AROC value for the current testing fold can be calculated.

- d. Repeat the above steps for the other $k-1$ sets of different training and testing folds and obtain the average AROC value from the k testing folds.
 - e. After obtaining the average AROC values for all possible combinations of $(\delta_i^1, \dots, \delta_i^m)$, identify the best combination $\delta^* = (\delta^{*1}, \dots, \delta^{*m})$ that corresponds to the highest average AROC value.
6. Construct the final Bayesian predictor by using the outcome table determined from the best combination of threshold $\delta^* = (\delta^{*1}, \dots, \delta^{*m})$ for the entire training peptides. That is, build the outcome table following the step 5.a with threshold $\delta^* = (\delta^{*1}, \dots, \delta^{*m})$ and the entire training set. Then compile the probability table as described in the above step 5.b. By varying threshold values for δ_B , obtain the corresponding sensitivity and specificity for the entire training set and compute an AROC value.

The general framework of building a Bayesian predictor is summarized in **Fig. 1**.

The threshold δ_b for the Bayesian classifier for testing has to be determined based on the requirement for sensitivity and specificity specified by users (*see Note 5*). The Bayesian predictor predicts a peptide as a binder if the highest value among the ratios $p(f = 1 | s_i) / p(f = 0 | s_i)$ for all overlapping 9-mers s_i from the peptide is great than δ_b ; otherwise predicts it as nonbinder.

For reference the performance of the Bayesian predictor built from the three individual predictors (i) ProPred (15), (ii) Gibbs sampler (16), and (iii) the LP predictor (17) in our illustrative example is shown in **Fig. 2**. The corresponding web server can be accessed at <http://array.bioengr.uic.edu/cgi-in/mhc2srv/testing.web.pl>.

4. Notes

1. In our study, peptide sequences were obtained from two databases: AntiJen (18) and MHCBN (19). Considering the size of training set, 9 alleles were selected: HLA-DRB1*0101, HLA-DRB1*0301, HLA-DRB1*0401, HLA-DRB1*0701, HLA-DRB1*0802, HLA-DRB1*1101, HLA-DRB1*1302, HLA-DRB1*1501, and HLA-DRB5*0101.
2. Each individual predictor may be a position specific scoring matrix, which is of size 20 by 9. The score of a 9-mer is defined as $\sum_{l=1}^9 s(l)$, where $s(l)$ is the value in the l th column of the matrix corresponding to the residue appeared at position l of the 9-mer. The score may not be the actual binding affinity of the 9-mer, however, the magnitude correlates the strength of the binding.

3. The AROC value is the area under receiver operating characteristic curve (20), which is determined from a set of values of (1-specificity, sensitivity) derived from different values of threshold of a predictor for a set of binder and nonbinder peptides. The sensitivity and specificity are defined as $TP/(TP+FN)$ and $TN/(TN+FP)$, respectively, where TP and FN are the respective numbers of predicted binders and nonbinders which are true binders; TN and FP are respective numbers of predicted nonbinders and binders which are true nonbinders. An AROC value close to 1 is desirable for a predictor. A random predictor has an AROC value of 0.
4. In the k -fold cross-validation, the ratio between the number of binders and the number of nonbinders in all k folds should be approximately equal. This is important for training.
5. The testing threshold δ_B for the Bayesian predictor is specified by the requirement of the users. In general, the recommended value for δ_B is that the sensitivity and specificity of the predictor are approximately equal. However, it is also possible to select a value for δ_B at which the sensitivity is higher than the specificity; or conversely, choose a value for δ_B at which the specificity is higher than the sensitivity. The values for δ_B and the corresponding values of the sensitivity and specificity can be obtained for the Bayesian predictor. These values indicate the quality that the predictor may have when the prediction is made for new peptide sequences. In our illustrative example, if one wishes the final predictor to target sensitivity at a level of 0.7, then the proper choice for δ_B should be 1.188 (*see Table 1*).

Acknowledgements

This work is partially supported by the NIH grant (1 R03 AI069391-01).

References

1. Flower, D. R. (2004) Vaccines in silico - the growth and power of immunoinformatics. *The Biochemist* **26** 17-20.
2. De Groot, A. S. and Berzofsky, J. A. (2004) From genome to vaccine--new immunoinformatics tools for vaccine design. *Methods* **34**, 425-428.
3. Parham, P. (2005) *The Immune System*. Garland Science, New York, NY.
4. Castellino, F., Zhong, G., and Germain, R. N. (1997) Antigen presentation by MHC class II molecules: invariant chain function, protein trafficking, and the molecular basis of diverse determinant capture. *Hum. Immunol.* **54**, 159-169.
5. Sette, A., Buus, S., Appella, E., Smith, J. A., Chesnut, R., Miles, C., Colon, S. M., and Grey, H. M. (1989) Prediction of major histocompatibility complex binding regions of protein antigens by sequence pattern analysis. *Proc. Natl. Acad. Sci. U. S. A.* **86**, 3296-3300.
6. Max, H., Halder, T., Kropshofer, H., Kalbus, M., Muller, C. A., and Kalbacher, H. (1993) Characterization of peptides bound to extracellular and intracellular HLA-DR1 molecules. *Hum. Immunol.* **38**, 193-200.
7. Rammensee, H., Bachmann, J., Emmerich, N. P., Bachor, O. A., and Stevanovic, S. (1999) SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* **50**, 213-219.

8. Borrás-Cuesta, F., Golvano, J., García-Granero, M., Sarobe, P., Riezu-Boj, J., Huarte, E., and Lasarte, J. (2000) Specific and general HLA-DR binding motifs: comparison of algorithms. *Hum. Immunol.* **61**, 266-278.
9. Sturniolo, T., Bono, E., Ding, J., Radrizzani, L., Tuereci, O., Sahin, U., Braxenthaler, M., Gallazzi, F., Protti, M. P., Sinigaglia, F., and Hammer, J. (1999) Generation of tissue-specific and promiscuous HLA ligand databases using DNA microarrays and virtual HLA class II matrices. *Nat. Biotechnol.* **17**, 555-561.
10. Kato, R., Noguchi, H., Honda, H., and Kobayashi, T. (2003) Hidden Markov model-based approach as the first screening of binding peptides that interact with MHC class II molecules. *Enzyme Microb. Technol.* **33**, 472-481.
11. Brusic, V., Rudy, G., Honeyman, G., Hammer, J., and Harrison, L. (1998) Prediction of MHC class II-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics* **14**, 121-130.
12. Nielsen, M., Lundegaard, C., Worning, P., Lauemoller, S. L., Lamberth, K., Buus, S., Brunak, S., and Lund, O. (2003) Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Sci.* **12**, 1007-1017.
13. Bhasin, M. and Raghava, G. P. (2004) SVM based method for predicting HLA-DRB1*0401 binding peptides in an antigen sequence. *Bioinformatics* **20**, 421-423.
14. Theodoridis, S. and Koutroumbas, K. (1999) *Pattern Recognition*. Academic Press, San Diego, CA.
15. Singh, H. and Raghava, G. P. (2001) ProPred: prediction of HLA-DR binding sites. *Bioinformatics* **17**, 1236-1237.

16. Nielsen, M., Lundegaard, C., Worning, P., Hvid, C. S., Lamberth, K., Buus, S., Brunak, S., and Lund, O. (2004) Improved prediction of MHC class I and class II epitopes using a novel Gibbs sampling approach. *Bioinformatics* **20**, 1388-1397.
17. Murugan, N. and Dai, Y. (2005) Prediction of MHC class II binding peptides based on an iterative learning model. *Immunome Res.* **1**, 6.
18. Toseland, C. P., Clayton, D. J., McSparron, H., Hemsley, S. L., Blythe, M. J., Paine, K., Doytchinova, I. A., Guan, P., Hattotuwigama, C. K., and Flower, D. R. (2005) AntiJen: a quantitative immunology database integrating functional, thermodynamic, kinetic, biophysical, and cellular data. *Immunome Res.* **1**, 4.
19. Bhasin, M., Singh, H., and Raghava, G. P. (2003) MHCBN: a comprehensive database of MHC binding and non-binding peptides. *Bioinformatics* **19**, 665-666.
20. Swets, J. A. (1988) Measuring the accuracy of diagnostic systems. *Science* **240**, 1285-1293.

Table1

Threshold values for the Bayesian predictor and the corresponding sensitivity and specificity for HLA-DRB1*0401 allele.

Threshold	Sensitivity	Specificity
1.602	0.000	1.000
1.601	0.554	0.876
1.188	0.710	0.817
0.849	0.790	0.774
0.630	0.842	0.651
0.563	0.878	0.505
0.402	0.918	0.339
0.299	1.000	0.000

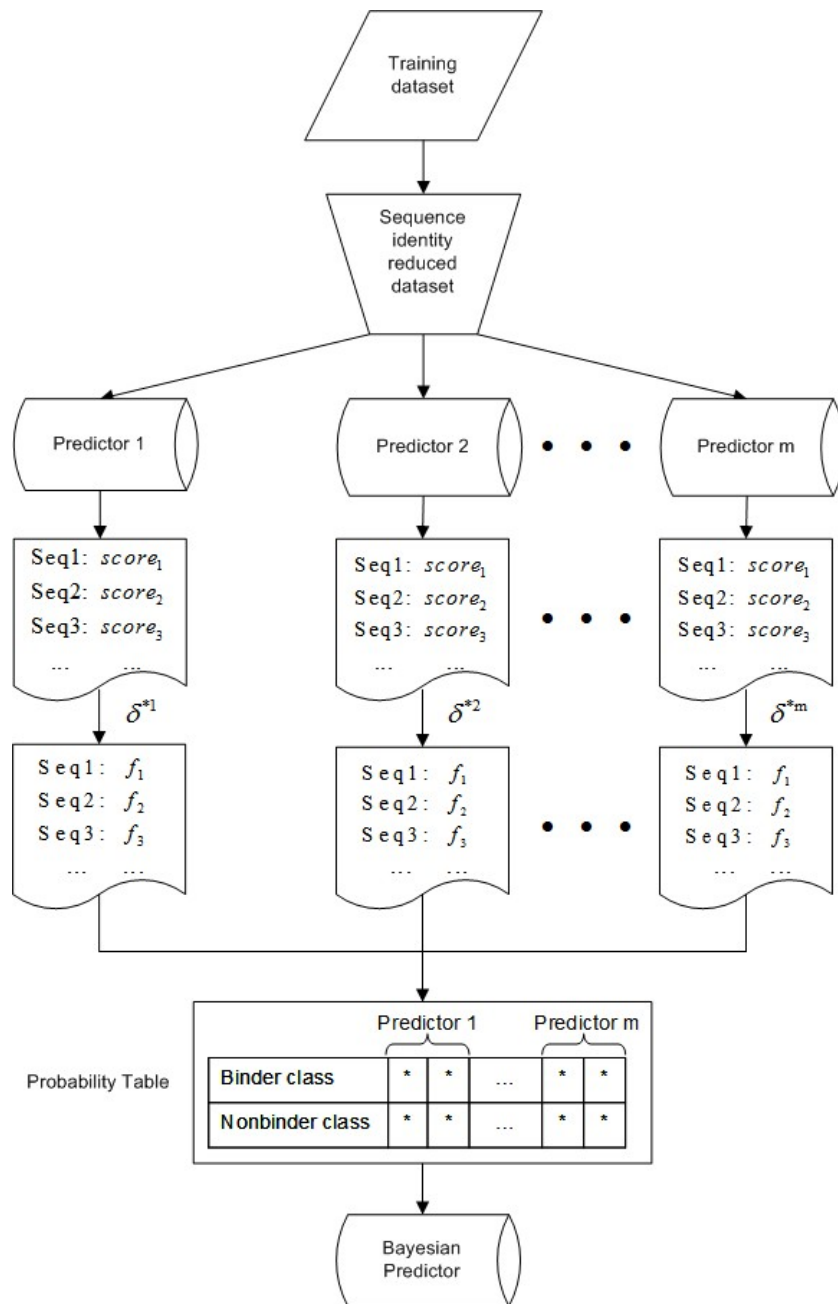


Fig. 1. Illustration of the framework for building a Bayesian predictor.

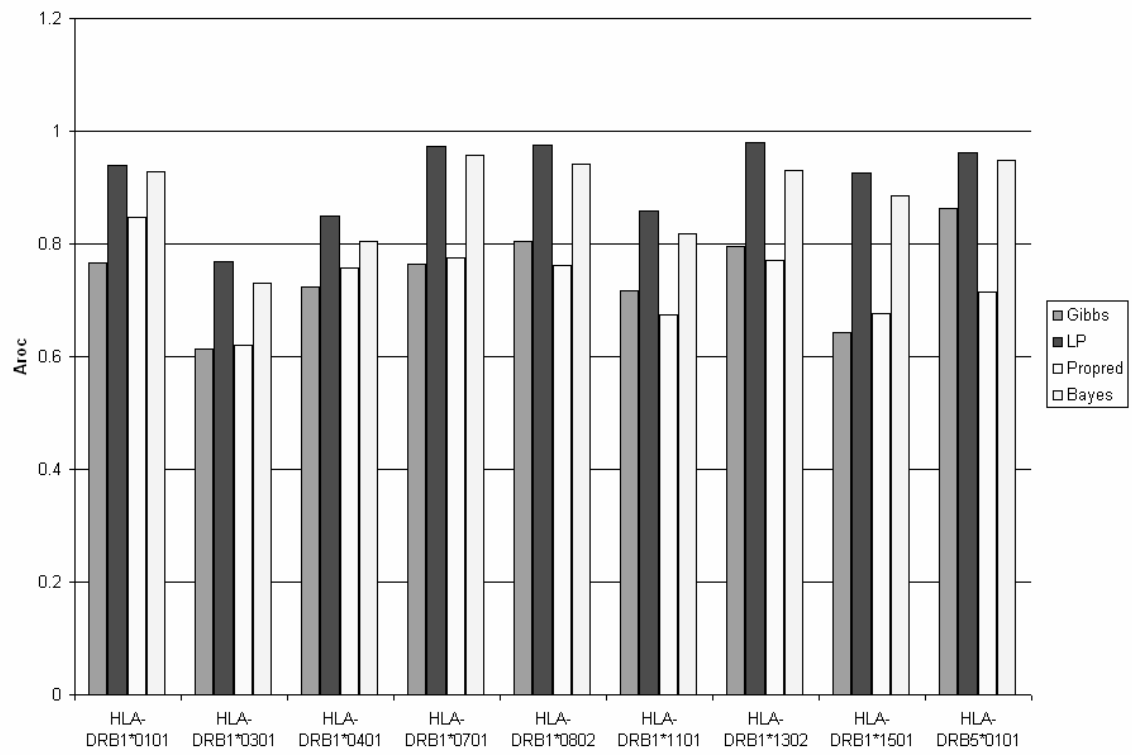


Fig. 2. Comparison of the performance of the Bayesian predictor with the three individual predictors.