Towards a Mathematical Foundation of Immunology and Amino Acid Chains*

Wen-Jun Shen[1], Hau-San Wong[1], Quan-Wu Xiao[2], Xin Guo[2], and Stephen Smale[2]

[1]Department of Computer Sciences, City University of Hong Kong
[2]Department of Mathematics, City University of Hong Kong

May 25, 2012

Abstract

We attempt to set a mathematical foundation of immunology and amino acid chains. To measure the similarities of these chains, a kernel on strings is defined using only the sequence of the chains and a good amino acid substitution matrix (e.g. BLOSUM62). The kernel is used in learning machines to predict binding affinities of peptides to human leukocyte antigens DR (HLA-DR) molecules. On both fixed allele [24] and pan-allele [23] benchmark databases, our algorithm achieves the state-of-the-art performance. The kernel is also used to define a distance on an HLA-DR allele set based on which a clustering analysis precisely recovers the serotype classifications assigned by WHO [14, 22]. These results suggest that our kernel relates well the chain structure of both peptides and HLA-DR molecules to their biological functions, and that it offers a simple, powerful and promising methodology to immunology and amino acid chain studies.

1 Introduction

Large scientific and industrial enterprises are engaged in efforts to produce new vaccines from synthetic peptides. The study of peptide binding to appropriate alleles is a major part of this effort. Our goal here is to support the use of a certain "string kernel" for peptide binding prediction as well for the classification of supertypes of the major histocompatibility complex (MHC, in humans which is also called HLA) alleles.

Our point of view, and our results imply, that some key biological information is contained in just two places: First in a similarity kernel (or substitution matrix) on the set of the fundamental amino acids; and second on a good representation of the relevant alleles as strings of these amino acids.

*The work described in this paper is supported by GRF grant [Project No. 9041544] and [Project No. CityU 103210] and [Project No. 9380050]. Email addresses: wenjshen@student.cityu.edu.hk (W.J. Shen), cshswong@cityu.edu.hk (H.S. Wong), qwxiao@live.com (Q.W. Xiao), xinguo2@cityu.edu.hk (X. Guo), and smale@cityu.edu.hk (S. Smale)

This is achieved with great simplicity and predictive power. Along the way we find that gaps and their penalties in the string kernels don't help, and that emphasizing peptide binding as a real-valued function rather than a binding/non-binding dichotomy clarifies the issues. We use a modification of BLOSUM62 followed by a Hadamard power. We also use regularized least squares in contrast to support vector machines as is consistent with our regression emphasis.

The construction (details later) of our main kernel $\hat{K}^3$ on amino acid chains, inspired by local alignment kernels (see e.g. [30]) as well as an analogous kernel in vision (see [38]) begins.

For purposes of this paper, a kernel K is a symmetric function $K : X \times X \to \mathbb{R}$ where X is a finite set. Given an order on X, K may be represented as a matrix (think of X as the set of indices of the matrix elements). Then it is assumed that K is positive definite (in such a representation).

Let $\mathscr{A}$ be the set of the 20 basic (for life) amino acids. Every protein has a representation as a string of elements of $\mathscr{A}$.

Step 1. Definition of a kernel $K^1 : \mathscr{A} \times \mathscr{A} \to \mathbb{R}$.

BLOSUM62 is a similarity (or substitution) matrix on $\mathscr{A}$ frequently used in immunology [13]. In the formulation of BLOSUM62, a kernel $Q : \mathscr{A} \times \mathscr{A} \to \mathbb{R}$ is defined using blocks of aligned strings of amino acids representing proteins. One can think Q as the "raw data" of BLOSUM62[1]. It is symmetric, positive-valued, and a probability measure on $\mathscr{A} \times \mathscr{A}$. (We have checked that it is positive definite.)

Let p be the marginal probability defined on $\mathscr{A}$ by Q. Thus

$$p(x) = \sum_{y \in \mathscr{A}} Q(x, y).$$

Next, we define the BLOSUM62-2 matrix, indexed by the set $\mathscr{A}$, as

$$[\text{BLOSUM62-2}](x, y) = \frac{Q(x, y)}{p(x)p(y)}.$$

We list the BLOSUM62-2 matrix in Appendix A. Suppose $\beta > 0$ is a parameter, usually chosen about $\frac{1}{8}$ or $\frac{1}{10}$ (still mysterious). Then a kernel $K^1 : \mathscr{A} \times \mathscr{A} \to \mathbb{R}$ is given by

$$K^1(x, y) = \left([\text{BLOSUM62-2}](x, y)\right)^{\beta}. \tag{1}$$

Note that the power in (1) is of the matrix entries, not of the matrix.

Step 2. Let $\mathscr{A}^1 = \mathscr{A}$ and define $\mathscr{A}^{k+1} = \mathscr{A}^k \times \mathscr{A}$ recursively for any $k \in \mathbb{N}$. We say s is an amino acid chain (or string) if $s \in \cup_{k=1}^{\infty} \mathscr{A}^k$, and $s = (s_1, \ldots, s_k)$ is a k-mer if $s \in \mathscr{A}^k$ for some $k \in \mathbb{N}$ with $s_i \in \mathscr{A}$. Consider

$$K_k^2(u, v) = \prod_{i=1}^{k} K^1(u_i, v_i)$$

where u, v are amino acid strings of the same length k, $u = (u_1, \ldots, u_k)$, $v = (v_1, \ldots, v_k)$; u, v are k-mers. K_k^2 is a kernel on the set of all k-mers.

[1] See Appendix A for the data.

Step 3. Let $f = (f_1, \cdots, f_m)$ be an amino acid chain. Denote $|f|$ as the length of f (so here $|f| = m$). Write $u \subset f$ whenever u is of the form $u = (f_{i+1}, \cdots, f_{i+k})$ for some $1 \leq i + 1 \leq i + k \leq m$. Let g be another amino acid chain, then define

$$K^3(f, g) = \sum_{\substack{u \subset f, v \subset g \\ |u|=|v|=k \\ \text{all } k=1,2,\ldots}} K_k^2(u, v).$$

Here and in all of this paper we abuse the notation a little bit to let the sum count each occurrence of u in f (and of v in g). While u and v need to have same length, not so for f and g. Replacing the sum by average gives a different but related kernel.

We define the correlation kernel $\hat{K}$ normalized from any kernel K by

$$\hat{K}(x, y) = \frac{K(x, y)}{\sqrt{K(x, x)K(y, y)}}.$$

In particular, let $\hat{K}^3$ be the correlation kernel of K^3.

Remark 1. *$\hat{K}^3$ is a kernel (see Section 2.2). It is symmetric, positive definite, positive-valued; it is basic for the results and development of this paper. We sometimes say string kernel. The construction works for any kernel (at the place of K^1) on any finite alphabet (replacing $\mathscr{A}$).*

Remark 2. *Comparison with the literature: See [12, 29, 31, 17]. But we use no gap penalty or even gaps, no logarithms, no implied round-offs, and no alignments (except the BLOSUM62-2 matrix which indirectly contains some alignment information). Our numerical experiments indicate that these don't help in our context, (at least!).*

Remark 3. *For complexity reasons one may limit the values of k in Step 3 with a small loss of accuracy, or even choose the k-mers at random.*

Remark 4. *The chains we use are proteins, peptides, and alleles. Peptides are short chain fragments of proteins, especially viruses and bacteria. Alleles are realizations of genes in living organisms varying with the individual; as proteins they have representations as amino acid chains.*

MHCII and MHCI are sets of alleles which are associated with immunological responses to viruses, bacteria, peptides and related. See [20, 10] for good introductions. In this paper we only study HLAII, the MHCII in human beings. HLA-DRB (or simply DRB) describes a subset of HLAII alleles which play a central role in immunology.

1.1 First Application: Binding Affinity Prediction

Peptide binding to a fixed HLAII (and HLAI as well) molecule (or an allele) a is a crucial step in the immune response of the human body to a pathogen or a peptide-based vaccine. Its prediction is computed from data of the form $(x_i, y_i)_{i=1}^m$, $x_i \in \mathscr{P}_a$ and $y_i \in [0, 1]$ where $\mathscr{P}_a$ is a set of peptides (i.e. chains of amino acids, in this paper we study peptides of length 9 to 37, usually about 15) associated to an HLAII allele a. The peptide binding problem occupies much research. We may use our kernel $\hat{K}^3$ described above for this problem since peptides are represented as strings of amino acids. Our prediction thus uses only the amino acid chains of the peptides, a substitution matrix, and some existing binding affinities (as “data”).

Following regularized least squares (RLS) supervised learning, the main construction is to compute

$$f_a = \arg \min_{f \in \mathscr{H}_{\hat{K}^3}} \sum_{i=1}^{m} (f(x_i) - y_i)^2 + \lambda \|f\|^2_{\hat{K}^3}. \tag{2}$$

Here $\lambda > 0$ and the index $\beta > 0$ in $\hat{K}^3$ are chosen by a procedure called leave-one-out cross validation. Also $\mathscr{H}_{\hat{K}^3}$ is the space of functions spanned by $\{\hat{K}^3_x : x \in \mathscr{P}\}$ (where $\hat{K}^3_x(y) := \hat{K}^3(x, y)$) on a finite set $\mathscr{P}$ of peptides containing $\mathscr{P}_a$. An inner product on $\mathscr{H}_{\hat{K}^3}$ is defined on the basis vectors by $\left\langle \hat{K}^3_x, \hat{K}^3_y \right\rangle_{\mathscr{H}_{\hat{K}^3}} = \hat{K}^3(x, y)$ by $\hat{K}^3$ above, then in general by linear extension. The norm of $f \in \mathscr{H}_{\hat{K}^3}$ is denoted by $\|f\|_{\hat{K}^3}$. f_a is the predicted peptide binding function. We refer to the algorithm as “KernelRLS”.

For the set of alleles, with the best data available we have Table 1. The area under the receiver operating characteristic curve (area under the ROC curve, AUC) is the main measure of accuracy used in the peptide binding literature. NN-W refers to the algorithm which up to now has achieved the most accurate results for this problem, although there are many previous contributions as [41, 18, 8]. In Section 2 there is more detail.

List of alleles, a	$\#\mathscr{P}_a$	KernelRLS		NN-W in [24]
		RMSE	AUC	AUC
DRB1*0101	5166	0.18660	**0.85707**	0.836
DRB1*0301	1020	0.18497	**0.82813**	0.816
DRB1*0401	1024	0.24055	**0.78431**	0.771
DRB1*0404	663	0.20702	0.81425	**0.818**
DRB1*0405	630	0.20069	**0.79296**	0.781
DRB1*0701	853	0.21944	0.83440	**0.841**
DRB1*0802	420	0.19666	**0.83538**	0.832
DRB1*0901	530	0.25398	**0.66591**	0.616
DRB1*1101	950	0.20776	**0.83703**	0.823
DRB1*1302	498	0.22569	0.80410	**0.831**
DRB1*1501	934	0.23268	**0.76436**	0.758
DRB3*0101	549	0.15945	0.80228	**0.844**
DRB4*0101	446	0.20809	0.81057	**0.811**
DRB5*0101	924	0.23038	**0.80568**	0.797
Average		0.21100	0.80260	0.798
Weighted Average		0.20451	0.82059	0.810

Table 1: The predicted performance of RLS on each fixed allele in the benchmark [24]. If a is the allele in column 1, then the number of peptides in $\mathscr{P}_a$ is given in column 2. The root-mean-square deviation (RMSE) scores are listed. The AUC scores of the RLS and the NN-W algorithm are listed for comparison, where a common threshold $\theta = 0.4256$ is used [24] in the final thresholding step into binding and non-binding (see Section 2.3 for the details). The weighted average scores are given by the weighting on the size $\#\mathscr{P}_a$ of the corresponding peptide sets $\mathscr{P}_a$. The best AUC in each row is marked in bold.

We remark on the simplicity and universality of the algorithm that is based on $\hat{K}^3$, which itself has this simplicity with the contributions from the substitution matrix (i.e. BLOSUM62-2) and the sequential representation of the peptides. There is an important generalization of the peptide binding problem where the allele is allowed to vary. Our results on this problem are detailed in Section 3.

1.2 Second Application: Clustering and Supertypes

We consider the classification problem of DRB (HLA-DR β chain) alleles into groups call supertypes as follows. The understanding of DRB similarities is very important for the designation of high population coverage vaccines. An HLA gene can generate a large number of allelic variants and this polymorphism guarantees a population from being eradicated by an individual pathogen. Furthermore, there are no more than twelve HLA II alleles in each individual [16] and each HLA II allele binds only to specific peptides [33, 43]. As a result, its difficult to design an effective vaccine for a large population. It has been demonstrated that many HLA molecules have overlapping peptide binding sets and there have been several attempts to group them into supertypes accordingly [36, 34, 37, 26, 19, 2, 4]. The supertypes are designed so that the HLA molecules in the same supertype will have a similar peptide binding specificity.

The Nomenclature Committee of the World Health Organization [22] has given extensive tables on serological type assignments to DRB alleles which are based on the work of many organizations and labs throughout the world. In particular the HLA dictionary 2008 by Holdsworth et al. [14] acknowledges especially data from the World Health Organization Nomenclature Committee for Factors of the HLA system, the International Cell Exchange and the National Marrow Donor Program. The text in Holdsworth et al., 2008 [14] indicates also the ambiguities of such assignments especially in certain serological types.

We define a set $\mathscr{N}$ of DRB alleles as follows. We downloaded 820 DRB allele sequences from the IMGT/HLA Sequence Database [27] [2]. And then 14 non-expressed alleles were excluded and there remain 806 alleles. For each allele, we only consider the amino acids located between the markers "RFL" (the location of the first occurrence of "RFL") and "TVQ" (the location of the last occurrence of "TVQ"). One reason is the majority of polymorphic positions occur in exon 2 of the HLA class II genes [11], and the amino acids located between the markers "RFL" and "TVQ" constitute the whole exon 2 [40]. The DRB alleles are encoded by 6 exons. Exon 2 is the most important component constituting an HLA II-peptide binding site. The other reason is in the HLA pseudo-sequences used in the NetMHCIIpan[25], all positions of the allele contacting with the peptide occur in this range. Thus each allele is transformed into a normal form. We should note that two different alleles may have the same normal form. For those alleles with the same normal form, we only consider the first one. The order is according to the official names of WHO. We collect the remaining 786 alleles with no duplicate normal forms into a set, call $\mathscr{N}$. This set not only includes all alleles listed in the tables of [14], but also contains some new alleles since 2008.

Thus $\mathscr{N}$ may be identified with a set of amino acid sequences. Next impose the kernel $\hat{K}^3$ above on $\mathscr{N}$ where $\beta = 0.06$, we call the kernel $\hat{K}^3_{\mathscr{N}}$.

On $\mathscr{N}$ we design a distance derived from $\hat{K}^3_{\mathscr{N}}$ by

$$D_{L^2}(a,b) = \left(\frac{1}{|\mathscr{N}|} \sum_{c \in \mathscr{N}} \left(\hat{K}^3_{\mathscr{N}}(a,c) - \hat{K}^3_{\mathscr{N}}(b,c) \right)^2 \right)^{1/2}, \qquad \forall a, b \in \mathscr{N}. \tag{3}$$

The DRB1*11 and DRB1*13 families of alleles have been the most difficult to deal with by the World Health Organization and for us as well. Therefore we will exclude the DRB1*11 and DRB1*13 families of alleles in the following cluster tree construction with the evidence that clustering of these 2 groups is ineffective. They are left to be analyzed separately.[3]

The set $\mathscr{M}$ consists of all DRB alleles except for the DRB1*11 and DRB1*13 families of alleles. $\mathscr{M}$ is a subset of the set $\mathscr{N}$. We produce a clustering of $\mathscr{M}$ based on the L^2 distance D_{L^2} restricted to $\mathscr{M}$, and use the OWA (Ordered Weighted Averaging) [42] based linkage instead of the "single" linkage in the hierarchical clustering algorithm.

This clustering uses no previous serological type information and no alignments. We have assigned supertypes labeled ST1, ST2, ST3, ST4, ST5, ST6, ST7, ST8, ST9, ST10, ST51, ST52 and ST53 to certain clusters in the Tree shown in Figure 1 based on contents of the bins described in Table 5. Peptides have played no role in our model. Differing from the artificial neural network method [21, 14], no training data" of any previously classified alleles are used in our clustering. We

[2] ftp://ftp.ebi.ac.uk/pub/databases/imgt/mhc/hla/DRB_prot.fasta

[3] We have found from a number of different experiments that "they do not cluster". Perhaps the geometric phenomenon here is in the higher dimensional scaled topology, i.e. the betti numbers $b_i > 0$, for $i > 0$.

make use of the DRB amino acid sequences to build the cluster tree. Only making use of these amino acid sequences, our supertypes are in exact agreement with the WHO assigned serological types [14], as can be seen by checking the supertypes against the bins.

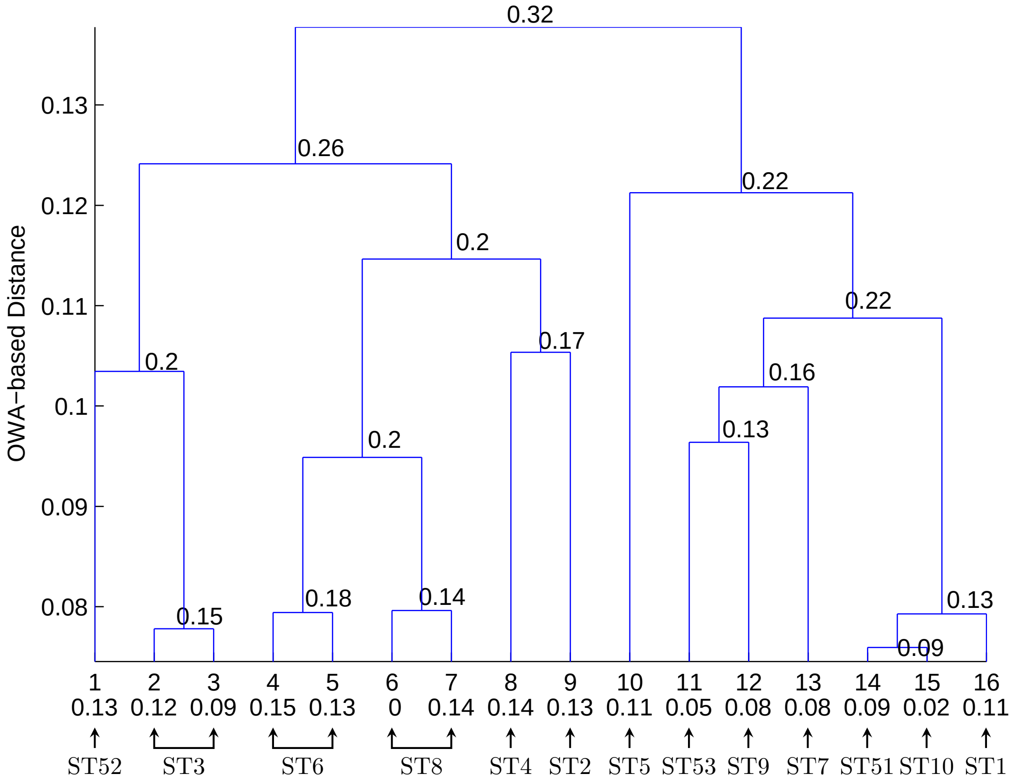

Figure 1: Cluster tree on 559 DRB alleles. The diameters of the leaf nodes are given at the bottom of the figure. The numbers given in the figure are the diameters of the corresponding unions of clusters.

This second application is given in some detail in Section 4.

2 Kernel Method for Binding Affinity Prediction

In this section we describe in detail the construction of our string kernel. The motivation is to relate the sequence information of strings (peptides or alleles) to their biological functions (binding affinities). A kernel works as a measure of similarity and supports the application of powerful machine learning algorithms such as regularized least squares (RLS) which we use in this paper. For a fixed allele, binding affinity is a function on peptides with values in $[0, 1]$. The function values on some peptides are available as the data, according to which RLS outputs a function that predicts for a new peptide the binding affinity to the allele. The method is generalized in the next section to the pan-allele kernel algorithm that takes also the allele structure into account.

2.1 Kernels

Suppose throughout the paper X is a finite set. We now give the definition of a kernel, of which an important example is our string kernel.

Definition 1. *A symmetric function $K : X \times X \to \mathbb{R}$ is called a kernel on X if it is positive definite, in the sense that by choosing an order on X, K can be represented as a positive definite matrix $(K(x, y))_{x,y\in X}$.*

Kernels have the following properties [5, 35, 1].

Lemma 1. *(i) If K is a kernel on X then it is also a kernel on any subset X_1 of X.*

(ii) If K_1 and K_2 are kernels on X, then $K : X \times X \to \mathbb{R}$ defined by

$$K(x, x') = K_1(x, x') + K_2(x, x')$$

is also a kernel.

(iii) If K_1 is a kernel on X_1 and K_2 is a kernel on X_2, then $K : (X_1 \times X_2) \times (X_1 \times X_2) \to \mathbb{R}$ defined by

$$K((x_1, x_2), (x_1', x_2')) = K_1(x_1, x_1') \cdot K_2(x_2, x_2')$$

is a kernel on $X_1 \times X_2$.

(iv) If K is a kernel on X, and f is a real-valued function on X that maps no point to zero, then $K' : X \times X$ defined by

$$K'(x, x') = f(x)K(x, x')f(x')$$

is also a kernel.

(v) If $K(x, x) > 0$ for all $x \in X$, then the correlation normalization of K given by

$$\hat{K}(x, x') = \frac{K(x, x')}{\sqrt{K(x, x)K(x', x')}} \tag{4}$$

is also a kernel.

Proof. (i), (ii) and (iv) follows the definition directly. (ii) follows the fact that the Kronecker product of two positive definite matrices is positive definite; see [15] for details. The positive definiteness of a kernel K guarantees that $K(x, x) > 0$ for any x in X, so (v) follows (iii). □

Remark 5. *Notice that with correlation normalization we have $\hat{K}(x, x) = 1$ for all $x \in X$. This is a desired property because the kernel function is usually used as a similarity measure, and with $\hat{K}$ we can say that each $x \in X$ is similar to itself.*

Define the real-valued function on X, K_x, by $K_x(y) = K(x, y)$. The function space $\mathscr{H}_K = \text{span}\{K_x : x \in X\}$ is a Euclidean space with inner product $\langle K_x, K_y\rangle = K(x, y)$, extended linearly to $\mathscr{H}_K$. The norm of functions in $\mathscr{H}_K$ is denoted as $\|\cdot\|_K$.

Remark 6. *The kernel can be defined even without assuming X is finite; in this general case the kernel is referred to as reproducing kernel [1]. If X is finite then a reproducing kernel is equivalent to our "kernel". The theory of reproducing kernel Hilbert spaces plays an important role in learning [5].*

On a finite set X there are two notions of distance derived from a kernel K. The first one is the usual distance in $\mathscr{H}_K$, that is

$$D_K(x, x') = \|K_x - K_{x'}\|_K,$$

for two points $x, x' \in X$. The second one is the L^2 distance defined by

$$D_{L^2}(x, x') = \left(\frac{1}{|X|} \sum_{t \in X} (K(x,t) - K(x',t))^2 \right)^{\frac{1}{2}}.$$

Important examples of the kernels discussed above are our kernel K^3 and its normalization $\hat{K}^3$.

2.2 Kernel on Strings

We start with a finite set $\mathscr{A}$ called the alphabet. In the work here $\mathscr{A}$ is the set of 20 amino acids, while the theory in this section applies to any other finite set. For example, as the name suggests, it can work on text for semantic analysis with a similar setting. See also [38] for the framework in vision.

To measure a similarity among the 20 amino acids, Henikoff and Henikoff [13] collect families of related proteins, align them and find conserved regions (i.e. regions that do not mutate frequently or greatly) as blocks in the families. The occurrence of each pair of amino acids in each column of every block is counted. A large number of occurrence indicates that in the conserved regions the corresponding pair of amino acids substitute each other frequently, or in another way of saying, they are similar. A symmetric matrix Q indexed by $\mathscr{A} \times \mathscr{A}$ is eventually obtained by normalizing the occurrences, so that $\sum_{x,y \in \mathscr{A}} Q(x,y) = 1$ and $Q(x,y)$ indicates the frequency of occurrences. See [13] for details, where the matrix Q is found from SCOP in this way, and the BLOSUM62 matrix is constructed accordingly.

Define $K^1 : \mathscr{A} \times \mathscr{A} \to \mathbb{R}$ as

$$K^1(x,y) = \left(\frac{Q(x,y)}{p(x)p(y)} \right)^{\beta}, \quad \text{depending on some } \beta > 0,$$

where

$$p(x) = \sum_{y \in \mathscr{A}} Q(x,y), \quad \forall x \in \mathscr{A},$$

is the marginal probability distribution on $\mathscr{A}$. When $\beta = 1$, we name the matrix $(K^1(x,y))_{x,y \in \mathscr{A}}$ as BLOSUM62-2 (one takes logarithm with base 2, scales it with factor 2, and rounds the obtained matrix to integers to obtain the BLOSUM62 matrix). Notice that if one chooses simply $Q = \frac{1}{m} I_{m \times m}$, then one obtains the matrix $I_{m \times m}$ as the analogue of the BLOSUM62-2, and the corresponding K^3 of the introduction is call the spectral kernel [17].

In the matrix language K^1 is the Hadamard power of the BLOSUM62-2 matrix, where for a matrix $M = (M_{i,j})$ with positive entries and a number $\beta > 0$, we denote $M^{\circ\beta}$ as the β'th Hadamard power of M and $\log^\circ M$ as the Hadamard logarithm of M, and their (i,j) entries are respectively,

$$(M^{\circ\beta})_{i,j} := (M_{i,j})^{\beta}, \qquad (\log^\circ M)_{i,j} := \log(M_{i,j}).$$

Theorem 1 (Horn and Johnson[15]). *Let A be an $m \times m$ positive-valued symmetric matrix. The Hadamard power $A^{\circ\beta}$ is positive definite for any $\beta > 0$ if and only if the Hadamard logarithm $\log^\circ A$ is conditionally positive definite (i.e. positive definite on the space $V = \{v = (v_1, \cdots, v_m) \in \mathbb{R}^m : \sum_{i=1}^m v_i = 0\}$).*

Proposition 1. *Every positive Hadamard power of BLOSUM62-2 is positive definite. Thus the above defined K^1 is a kernel.*

Proof. One just shows the eigenvalues of the Hadamard logarithm on V are all positive. One checks this by computer.

Theorem 2. *Based on any kernel K^1, the functions K^2_k, K^3, and $\hat{K}^3$ defined as in the introduction are all kernels.*

Proof. The fact K^2_k is a kernel for $k \geq 1$ follows Lemma 1 (iii). We now prove that K^3 is positive definite on any finite set X of strings, which then implies the positive definiteness of $\hat{K}^3$ by Lemma 1 (v). From Lemma 1 (i) it suffices to verify the cases that $X = X_k = \cup_{i=1}^k \mathscr{A}^i$ for $k \geq 1$. When $k = 1$, K^3 is just K^1 and hence positive definite. We assume now that K^3 is positive definite on X_k with $k = n$.

We claim that the matrices indexed by X_{n+1},

$$K^3_{i,X_{n+1}}(f,g) = \begin{cases} \sum_{\substack{u \subset f, v \subset g \\ |u|=|v|=i}} K^2(u,v) & \text{if } |f|, |g| \geq i, \\ 0 & \text{if } |f| < i \text{ or } |g| < i, \end{cases}$$

are all positive semi-definite. In fact, for any $1 \leq i \leq n$,

$$K^3_{i,X_{n+1}} = P_i K^2_i P_i^T, \tag{5}$$

where K^2_i is the matrix $(K^2_i(u,v))_{u,v \in \mathscr{A}^i}$, and P_i is a matrix with X_{n+1} as the row index set and $\mathscr{A}^i$ as the column index set, and for any $f \in X_{n+1}$ and $u \in \mathscr{A}^i$, $P_i(f,u)$ counts the times u occurs in f. Let us explain equation (5) a little more. For f and g in X_{n+1}, from the definition of P_i we have

$$(P_i K^2_i P_i^T)(f,g) = \sum_{u,v \in \mathscr{A}^i} P_i(f,u) P_i(g,v) K^2_i(u,v) = \sum_{\substack{u \subset f, v \subset g \\ |u|=|v|=i}} K^2_i(u,v), \quad \forall i. \tag{6}$$

Summing the equation (6) above over i gives the definition of $K^3(f,g)$.

For $i = n+1$, we have

$$K^3_{n+1,X_{n+1}}(f,g) = \begin{cases} 0 & f \notin \mathscr{A}^{n+1} \text{ or } g \notin \mathscr{A}^{n+1}, \\ K^2_{n+1}(f,g) & \text{otherwise.} \end{cases}$$

Therefore $K^3_{n+1,X_{n+1}}$ is positive definite on $\mathscr{A}^{n+1}$, and is zero elsewhere. Since

$$K^3(f,g) = \sum_{i=1}^{n} K^3_{i,X_{n+1}}(f,g), \quad \forall f, g \in X_n,$$

we know that the sum of $K^3_{i,X_{n+1}}$ with $i = 1, \cdots, n$ are positive definite on X_n, and positive semi-definite on X_{n+1}. Because

$$K^3(f,g) = \sum_{i=1}^{n+1} K^3_{i,X_{n+1}}(f,g), \quad \forall f, g \in X_{n+1},$$

we see that K^3 is positive definite on X_{n+1}. □

Corollary 1. *Our kernels K^2_k, K^3 and $\hat{K}^3$ are discriminative, that is, given any two strings f, g in the domain of K, as long as $f \neq g$, $D_K(f,g) > 0$. Here K stands for the three kernels.*

2.3 First Application: Peptide Affinities Prediction

We first briefly review the RLS algorithm inspired by learning theory. Let K be a kernel on a finite set X. Write $\mathscr{H}_K$ to denote the inner product space of functions on X defined by K. Suppose $\bar{z} = \{(x_i, y_i)\}_{i=1}^m$ is a sample set with $x_i \in X$ and $y_i \in \mathbb{R}$ for each i. The RLS uses a positive parameter $\lambda > 0$ and $\bar{z}$ to generate the output function $f_{\bar{z},\lambda} : X \to \mathbb{R}$,

$$f_{\bar{z},\lambda} = \arg \min_{f \in \mathscr{H}_K} \left\{ \frac{1}{\#\bar{z}} \sum_{(x_i,y_i)\in\bar{z}} \left(f(x_i) - y_i\right)^2 + \lambda \|f\|_K^2 \right\}. \tag{7}$$

Since $\mathscr{H}_K$ is of finite dimension, one solves (7) by representing f linearly by functions K_x with $x \in X$ and finding the coefficients. See [5, 32] for details.

Remark 7. *The RLS algorithm (7) is independent of the choice of the underlying space X where the function space $\mathscr{H}_K$ is defined, in the sense that the predicted values $f_{\bar{z},\lambda}(x)$ at $x \in X$ will not be changed if we extend K onto a large set $X' \supset X$ and re-run (7) with the same $\bar{z}$ and λ. This is guaranteed by the construction of the solution. See, e.g. [5, 32].*

Leave-one-out cross validation is employed to find the parameter λ in this paper. First, one gives a candidate set Λ of λ. For each $(x_i, y_i) \in \bar{z}$, one denotes $\bar{z}^i$ as the new dataset obtained by removing (x_i, y_i) from $\bar{z}$. One applies RLS on $\lambda \in \Lambda$ and $\bar{z}^i$ to obtain the predicted value $f_{\bar{z}^i,\lambda}(x_i)$ on x_i. The parameter $\lambda \in \Lambda$ is chosen so that the predicted error $\sum_{i=1}^m \left(f_{\bar{z}^i,\lambda}(x_i) - y_i\right)^2$ is minimized.

Binding affinity measures the strength that a peptide binds to an allele with, and is represented by IC50 score. Usually an IC50 score lies between 0 and 50,000 (nano molar). A widely used IC50 threshold determining binding and non-binding is 500 ("binding" if the IC50 value is less than 500). The bioinformatics community usually normalize the scores by the function $\psi_b : (0, +\infty) \to [0, 1]$ with a base $b > 1$,

$$\psi_b(x) := \begin{cases} 0 & x > b, \\ 1 - \log_b x & 1 \le x \le b, \\ 1 & x \le 1. \end{cases} \tag{8}$$

Without introducing any ambiguity we will in the sequel refer to the normalized IC50 value as the biding affinity.

We test the kernel with RLS on the IEDB benchmark data set published on [25]. The data set covers 14 DRB alleles, each allele a with a set $\mathscr{P}_a$ of peptides. For any $p \in \mathscr{P}_a$, its sequence representation and the $[0, 1]$-valued binding affinity $y_{a,p}$ to the allele a are both given. On the data set we compare our algorithm with the state-of-the-art NN-align algorithm [24]. In [24] for each allele a, the peptide set $\mathscr{P}_a$ was divided into 5 parts for validating the performance[4].

Now fix an allele a. Set $X = \mathscr{P} \supset \mathscr{P}_a$ (Remark 7 shows that one may select any finite $\mathscr{P}$ that contains $\mathscr{P}_a$ here). Define the kernel $\hat{K}^3$ on X through the steps in the Introduction (leaving the power index β to be fixed). We use the same 5-fold partition $\mathscr{P}_a = \cup_{t=1}^5 \mathscr{P}_{a,t}$ as in [25], and test five times the algorithm (7) with $K = \hat{K}^3$. In the t'th test ($t = 1, \cdots, 5$) four parts of $\mathscr{P}_a$ are merged to

[4] Both the data set and the 5-fold partition are available at `http://www.cbs.dtu.dk/suppl/immunology/NetMHCII-2.0.php`.

be the training data, denoted as $\mathscr{P}_a^{(t)} = \mathscr{P}_a \backslash \mathscr{P}_{a,t}$, and $\mathscr{P}_{a,t}$ is left as the testing data. We determine the parameter β in $\hat{K}^3$ and the regularization parameter λ in (7) by leave-one-out cross validation with $\bar{z} = \mathscr{P}_a^{(t)}$. Every pair of β in the geometric sequence $\{0.001, \cdots, 10\}$ of length 30 and λ in the geometric sequence $\{\mathrm{e}^{-17}, \cdots, \mathrm{e}^{-3}\}$ of length 15 is tested. With the optimal pair $(\beta_a^{(t)}, \lambda_a^{(t)})$, we train the RLS (7) once more on $\mathscr{P}_a^{(t)}$ to give the predicted binding function $f_{\mathscr{P}_a^{(t)}, \lambda_a^{(t)}, \beta_a^{(t)}}$ on $\mathscr{P}$. After the five times of testing on allele a, we denote $\tilde{y}_{a,p} = f_{\mathscr{P}_a^{(t)}, \lambda_a^{(t)}, \beta_a^{(t)}}(p)$ for each $p \in \mathscr{P}_{a,t}$ and $t = 1, \cdots, 5$.

The RMSE score is evaluated as

$$RMSE_a = \sqrt{\frac{1}{\#\mathscr{P}_a} \sum_{p \in \mathscr{P}_a} \left(\tilde{y}_{a,p} - y_{a,p}\right)^2}.$$

A smaller RMSE score indicates a better predicting performance. Since the affinity labels in this data set are transformed with $\psi_{b=50{,}000}$, there is a threshold $\theta = \psi_{50{,}000}(500) \approx 0.4256$ in [24] dividing a peptide $p \in \mathscr{P}_a$ into "binding" if $y_{a,p} > \theta$ and "non-binding" otherwise, to the allele a. Denote $\mathscr{P}_{a,B} = \{p \in \mathscr{P}_a : y_{a,p} > \theta\}$ and $\mathscr{P}_{a,N} = \mathscr{P}_a \backslash \mathscr{P}_{a,B}$, then the AUC index is evaluated as

$$AUC_a = \frac{\#\{(p, p') : p \in \mathscr{P}_{a,B},\, p' \in \mathscr{P}_{a,N},\, \tilde{y}_{a,p} > \tilde{y}_{a,p'}\}}{(\#\mathscr{P}_{a,B})\,(\#\mathscr{P}_{a,N})}. \tag{9}$$

The sequence of ideas for each allele a leads to Table 1 The computation also suggests a weighted optimal index

$$\beta^*_{peptide} := \frac{1}{\sum_a \#\mathscr{P}_a} \sum_a \left\{ (\#\mathscr{P}_a) \left(\frac{1}{5} \sum_{t=1}^{5} \beta_a^{(t)} \right) \right\} = 0.11387. \tag{10}$$

We will use this value in the next section.

Remark 8. *We take the point of view that peptide binding is a matter of degree and hence is better measured by a real number, rather than the binding–non-binding dichotomy. Thus RMSE is a better measure than AUC. The results in Table 1 also demonstrate that the regression-base learning model works well on the real-valued data.*

Remark 9. *Our philosophy is that there is a metric structure on the set of amino acid sequences related to their biological functions (e.g. the distances on peptides related to their affinities to for each allele). The metric should not depend heavily on the alignment information, which is possibly a big source of noise. The performance of our kernel $\hat{K}^3$ is reflected by the modulus of continuity of the prediction values, namely,*

$$\Omega_a := \max_{p, p' \in \mathscr{P}_a} \frac{|\tilde{y}_{a,p} - \tilde{y}_{a,p'}|}{d(p, p')},$$

where

$$d(p, p') = \|\hat{K}^3_p - \hat{K}^3_{p'}\|_{\hat{K}^3} = \sqrt{2 - 2\hat{K}^3(p, p')},$$

which is the distance in the space $\mathscr{H}_{\hat{K}^3}$ on peptides, and the kernel $\hat{K}^3$ is defined with $\beta = \beta^*_{peptide}$. We list the values of Ω_a for the 14 alleles in Table 2.

Allele a	Ω_a	Allele a	Ω_a	Allele a	Ω_a
DRB1*0101	1.2222	DRB1*0301	1.0307	DRB1*0401	0.9249
DRB1*0404	0.9726	DRB1*0405	0.8394	DRB1*0701	1.1317
DRB1*0802	0.9368	DRB1*0901	0.8004	DRB1*1101	0.9795
DRB1*1302	0.7745	DRB1*1501	0.9843	DRB3*0101	0.7395
DRB4*0101	0.8587	DRB5*0101	1.0011		

Table 2: The module of continuity of the prediction values.

The modulus of continuity can be extended to a bigger peptide set $\mathscr{P}'$ which contains the neighbourhood of each peptide $p \in \mathscr{P}$ with respect to the metric d.

3 Kernel Algorithm for pan-Allele Binding Prediction

We now define a pan-allele kernel on the product space of alleles and peptides. The binding affinity data is thus a subset of this product space. The main motivation is that by the pan-allele kernel we predict affinities to those alleles with few or no binding data available: this is often the case because the MHCII alleles form a huge set (the phenomenon is often referred to as MHCII polymorphism), and the job determining experimentally peptide affinities to all the alleles is immense. Also, in the pan-allele setting, one puts the binding data to different alleles together to train the RLS. This makes the training data set larger than that was available in the fixed allele setting, and thus helps to improve the prediction performance. This is verified in Table 4.

Let $\mathscr{L}$ be a finite set of amino acid sequences representing the MHC2 alleles. Using a positive parameter β_{allele} we define a kernel $\hat{K}^3_{\mathscr{L}}$ on $\mathscr{L}$ following the steps in the Introduction. Let $\mathscr{P}$ be a set of peptides. In the sequel we denote by $\beta_{peptide}$ specifically the parameter used to define the kernel $\hat{K}^3_{\mathscr{P}}$ on $\mathscr{P}$. We define the pan-allele kernel on $\mathscr{L} \times \mathscr{P}$ as

$$\hat{K}^3_{pan}((a,p),(a',p')) = \hat{K}^3_{\mathscr{L}}(a,a')\hat{K}^3_{\mathscr{P}}(p,p'). \tag{11}$$

Let be given a set of data $\{(p_i, a_i, r_i)\}_{i=1}^m$. Then for each i, $a_i \in \mathscr{L}$, $p_i \in \mathscr{P}$, and $r_i \in [0,1]$ is the binding affinity of p_i to a_i. The RLS is applied as in Section 2. The output function $F : \mathscr{L} \times \mathscr{P} \to \mathbb{R}$ is the predicted the binding affinity.

Remark 10. *When we choose $\mathscr{L} = \{a\}$ for a certain allele a, the setting and the algorithm reduce to the fixed-allele version studied in Section 2.*

We test the pan-allele kernel with RLS (we call the algorithm "KernelRLSpan") on Nielsen's NetMHCIIpan-2.0 data set (we also denote by the name the algorithm published on [23] with the data set), which contains 33,931 peptide-allele pairs. For peptides, amino acid sequences are given, and for alleles, DRB names are given so that we can find out the sequence representation in $\mathscr{N}$ as defined in Section 1.2. Each pair is labeled with a $[0,1]$-valued binding affinity. These peptide-allele pairs cover 24 alleles in $\mathscr{N}$ and 8083 peptides. The whole data set is divided into 5 parts in [23][5]

[5] Both the data set and the 5-part partition are available at `http://www.cbs.dtu.dk/suppl/immunology/NetMHCIIpan-2.0`.

We choose this setting. Let $\mathscr{L} = \mathscr{N}$ and $\mathscr{P}$ be a peptide set large enough to contain all the peptides in the data set. We use $\beta^*_{peptide} = 0.11387$ as suggested in (10) to construct $\hat{K}^3_{\mathscr{P}}$ and leave the power index β_{allele} for $\hat{K}^3_{\mathscr{N}}$ to be fixed later. This defines $\hat{K}^3_{pan}$. We test the RLS algorithm 5 times according to the 5-part division in [23]. In each test we merge 4 parts of the samples as the training data and leave the other part as the testing data. Leave-one-out cross validation is employed in each test and we choose a pair $(\beta_{allele}, \lambda)$ from the product of $\{0.02 \times n : n = 1, 2, \cdots, 8\}$ and $\{\mathrm{e}^n : n = -17, -16, \cdots, -9\}$. The procedures are the same as used in Section 2.3 except we now do cross validation for the peptide-allele pairs. In all the tests, the pair $\beta_{allele} = 0.06$ and $\lambda = \mathrm{e}^{-13}$ achieves the best performance in the training data. We now use the threshold $\theta = \psi_{15,000}(500) \approx 0.3537$ to evaluate the AUC score, because the affinity values in the data set is obtained by the transform $\psi_{15,000}$. the ideas lead to Table 3

allele, a	$\#\mathscr{P}_a$	KernelRLS		NetMHCIIpan-2.0
		RMSE	AUC	AUC
DRB1*0101	7685	0.20575	0.84308	**0.846**
DRB1*0301	2505	0.18154	0.85095	**0.864**
DRB1*0302	148	0.21957	0.71176	**0.757**
DRB1*0401	3116	0.19860	0.84294	**0.848**
DRB1*0404	577	0.21887	0.80931	**0.818**
DRB1*0405	1582	0.17459	**0.86862**	0.858
DRB1*0701	1745	0.17769	**0.87664**	0.864
DRB1*0802	1520	0.18732	**0.78937**	0.780
DRB1*0806	118	0.23091	0.89214	**0.924**
DRB1*0813	1370	0.18132	**0.88803**	0.885
DRB1*0819	116	0.18823	**0.82706**	0.808
DRB1*0901	1520	0.19741	**0.82220**	0.818
DRB1*1101	1794	0.16022	**0.88610**	0.883
DRB1*1201	117	0.22740	0.87380	**0.892**
DRB1*1202	117	0.23322	0.89440	**0.900**
DRB1*1302	1580	0.19953	0.82298	**0.825**
DRB1*1402	118	0.20715	**0.86474**	0.860
DRB1*1404	30	0.18705	0.64732	**0.737**
DRB1*1412	116	0.26671	**0.89967**	0.894
DRB1*1501	1769	0.19609	**0.82858**	0.819
DRB3*0101	1501	0.15271	0.82921	**0.85**
DRB3*0301	160	0.26467	**0.86857**	0.853
DRB4*0101	1521	0.16355	**0.87138**	0.837
DRB5*0101	3106	0.18833	0.87720	**0.882**
Average		0.20035	0.84109	0.846
W. Average		0.19015	0.84887	0.849

Table 3: The performance of KernelRLSpan. For comparison we list the AUC scores of NetMHCIIpan-2.0 [23]. The weighted average values are given by the weighting on the size of the corresponding peptide sets. The best AUC in each row are marked in bold.

We implement KernelRLSpan on the fixed allele data set used in Table 1. Recall that the data set is normalized with $\psi_{50,000}$ and has the 5-fold division defined by [25]. The performance is listed in Table 4, which is better than that of KernelRLS as listed in Table 1.

allele, a	RMSE	AUC	allele, a	RMSE	AUC
DRB1*0101	0.17650	0.86961	DRB1*0301	0.16984	0.85601
DRB1*0401	0.20970	0.82359	DRB1*0404	0.17240	0.88193
DRB1*0405	0.18425	0.84078	DRB1*0701	0.17998	0.90231
DRB1*0802	0.16734	0.88496	DRB1*0901	0.23562	0.71057
DRB1*1101	0.17073	0.91022	DRB1*1302	0.23261	0.75960
DRB1*1501	0.21266	0.80724	DRB3*0101	0.16011	0.79778
DRB4*0101	0.18751	0.84754	DRB5*0101	0.18904	0.89585
Average:	RMSE 0.18916, AUC 0.84200				
Weighted Average:	RMSE 0.18496, AUC 0.85452				

Table 4: The performance of KernelRLSpan on the fixed allele data. For defining AUC, the transform $\psi_{50,000}$ is used as in Table 1.

Next, we use the whole NetMHCIIpan-2.0 data set for training, and test the prediction performance on a new data set. A set of 64798 pieces of peptide affinity data is downloaded from IEDB[6]. We pick from the set the data that are about DRB alleles, having IC50 scores, and having explicit allele name and peptide sequence. Those pieces also appear in the NetMHCIIpan-2.0 data set are deleted. For the duplicated pieces (same peptide-allele pair and same affinity) only one of them are kept. All the pieces of the same peptide-allele pair yet gave different affinities are deleted. We deleted those with peptide length less than 9. (The KernelRLSpan itself can handle these peptides, while the NetMHCIIpan-2.0 cannot. The short peptides are deleted to make the two algorithms comparable.) For some alleles the data in the set is insificient to define the AUC score (i.e. the denominator in (9) is made zero), so we delete all the pieces related to them. Eventually we obtained 11334 peptide-allele pairs labelled with IC50 binding affinities, which are further normalized by $\psi_{15,000}$ as in the NetMHCIIpan-2.0 data set. Now define $\hat{K}^3_{pan}$ on $\mathscr{N} \times \mathscr{P}$ as in (11) with $\beta_{allele} = 0.06$ as suggested by the above computation and $\beta_{peptide} = 0.11387$ as suggested in (10). We train on the NetMHCIIpan-2.0 data set both KernelRLSpan and NetMHCIIpan-2.0[7]. In the KernelRLSpan, leave-one-out cross validation is used to select λ from $\{e^{-18}, \cdots, e^{-8}\}$ (the result shows that $\lambda = e^{-13}$ performs the best). The prediction performance of the two algorithms are compared on Table 5.

[6] The data set was downloaded from http://www.immuneepitope.org/list_page.php?list_type=mhc&measured_response=&total_rows=64797&queryType=true, on May 23, 2012.

[7] The code is published on http://www.cbs.dtu.dk/cgi-bin/nph-sw_request?netMHCIIpan.

allele, a	$\#\mathscr{P}_a$	kernelRLSpan		NetMHCIIpan-2.0	
		RMSE	AUC	RMSE	AUC
DRB1*0101	1024	0.25519	0.79717	**0.24726**	**0.82988**
DRB1*0102	7	**0.39748**	**0.58333**	0.62935	**0.58333**
DRB1*0103	41	0.33159	**0.83333**	**0.32204**	**0.83333**
DRB1*0301	883	**0.21760**	0.80276	0.23975	**0.82384**
DRB1*0401	1122	0.19610	0.79930	**0.19363**	**0.82456**
DRB1*0402	48	**0.23912**	**0.67321**	0.27352	0.65714
DRB1*0403	43	0.16381	**0.70443**	**0.15868**	0.66995
DRB1*0404	494	0.21689	0.79344	**0.20219**	**0.82517**
DRB1*0405	462	0.19617	0.78941	**0.19387**	**0.80611**
DRB1*0406	14	0.19516	0.53846	**0.19497**	**0.61538**
DRB1*0701	724	0.20853	0.80876	**0.20039**	**0.84786**
DRB1*0801	24	0.37281	**0.72500**	**0.34767**	0.71250
DRB1*0802	404	0.17403	0.80407	**0.17181**	**0.81085**
DRB1*0901	335	0.21204	0.79524	**0.21029**	**0.80489**
DRB1*1001	20	0.28082	0.74000	**0.24335**	**0.92000**
DRB1*1101	811	0.24195	0.83219	**0.23838**	**0.85071**
DRB1*1104	10	**0.43717**	**0.76190**	0.57082	0.57143
DRB1*1201	795	0.25786	**0.83178**	**0.24984**	0.82685
DRB1*1301	147	**0.27014**	0.65077	0.30202	**0.70722**
DRB1*1302	499	0.22194	0.82118	**0.21284**	**0.84258**
DRB1*1501	856	0.21580	0.83563	**0.20869**	**0.84902**
DRB1*1502	3	**0.13186**	**1.00000**	0.20061	**1.00000**
DRB1*1601	16	0.19556	**0.84615**	**0.18740**	0.76923
DRB1*1602	12	0.32238	**0.68571**	**0.30431**	0.60000
DRB3*0101	437	**0.16568**	0.74058	0.17860	**0.77182**
DRB3*0202	750	**0.16021**	0.82543	0.16453	**0.84191**
DRB4*0101	563	**0.20594**	**0.80575**	0.21383	0.78734
DRB5*0101	774	0.25934	0.78701	**0.25849**	**0.81950**
DRB5*0202	16	**0.23013**	**0.71429**	0.40554	0.57143
Average		**0.24046**	0.76987	0.25947	**0.77151**
Wtd. Ave.		0.21853	0.80309	**0.21816**	**0.82216**

Table 5: The performance of KernelRLSpan and NetMHCIIpan-2.0 trained on the NetMHCIIpan-2.0 benchmark data set, tested on a new dataset downloaded from the IEDB. The best performance of both AUC and RMSE scores of each row is marked in bold.

In this section KernelRLSpan is tested. The results suggests that compared with KernelRLS, KernelRLSpan performs much better. Also, the kernel method uses only the substitution matrix and the sequence representations without direct alignment information but yields comparable performance with the state-of-the-art NetMHCIIpan-2.0 algorithm.

4 Clustering and Supertypes

In this section, we describe in detail the construction of our cluster tree and our classification of DRB alleles into supertypes. We compare the supertypes identified by our model with the serotypes designated by WHO (World Health Organization) and analyze the comparison results in detail.

4.1 Identification of DRB Supertypes

We classify DRB alleles into disjoint subsets by the use of DRB amino acid sequences and the BLOSUM62 substitution matrix. No peptide binding data or X-ray 3D structure data are used in our clustering. We obtain a classification in this way into subsets (a partition) which we call supertypes.

In section 3, we have defined the allele kernel on $\mathscr{N}$ as $\hat{K}^3_{\mathscr{N}}$; the L^2 distance derived from $\hat{K}^3_{\mathscr{N}}$ is defined as

$$D_{L^2}(x,y) = \left(\frac{1}{|\mathscr{N}|} \sum_{z \in \mathscr{N}} \left(\hat{K}^3(x,z) - \hat{K}^3(y,z) \right)^2 \right)^{1/2}, \qquad \forall x, y \in \mathscr{N}.$$

The OWA-based linkage is used to measure the proximity between clusters X and Y [8]. Let $U = (d_{xy})_{x \in X, y \in Y}$, where $d_{xy} = D_{L^2}(x,y)$. After ordering the elements of U in descending order, we obtain an ordered vector $V = (d'_1, \ldots, d'_n), n = |U|$. A weighting vector $W = (w_1, \cdots, w_n)$ is associated with V, and the proximity between clusters X and Y is defined as

$$D_{OWA}(X,Y) = \sum_{i=1}^{n} w_i d'_i.$$

Here the OWA weights W are defined as follows [28]:

$$\begin{aligned} w'_i &= \frac{\mathrm{e}^{i/\mu}}{\mu}, \quad i = 1, 2, \cdots, n, \\ w_i &= \frac{w'_i}{\sum_{j=1}^{n} w'_j}, \quad i = 1, 2, \cdots, n, \end{aligned}$$

where $\mu = \gamma(1+n)$, γ is chosen appropriately as 0.1. This weighting gives more importance to pairs (x,y) which have smaller distance.

Hierarchical clustering [6] is applied to build a cluster tree. A cluster tree is a tree and every node in a cluster tree represents a cluster as the set of all leaves descended from that node. The L_2 distance D_{L^2} is used to measure the distance between alleles x and y, $x, y \in \mathscr{M}$ and OWA-based linkage is used to measure the proximity between clusters X and Y, $X, Y \subseteq \mathscr{M}$ instead of "single" linkage. This algorithm is a bottom-up approach. At the beginning, each allele is treated as a singleton cluster, and then successively merge two nearest clusters X and Y into a union cluster, this process will stop until all unions of clusters have been merged into a single cluster.

This cluster tree, associated to $\mathscr{M}$, has thus 559 leaves. The upper part of this tree is shown in Figure 1. We assign supertypes to certain clusters in the cluster tree based on contents of the bins described in Table 5. Thirteen supertypes are defined in this way, which we name ST1, ST2, ST3, ST4, ST5, ST6, ST7, ST8, ST9, ST10, ST51, ST52 and ST53. The corresponding cluster diameters are 0.11, 0.13, 0.15, 0.14, 0.11, 0.18, 0.08, 0.14, 0.08, 0.02, 0.09, 0.13 and 0.05, respectively.

The diameter of a cluster Z is defined as

$$diameter(Z) = \max_{x,y \in Z} D_{L^2}(x,y). \tag{12}$$

[8] Another good way of measuring distance between clusters is the Hausdorff distance.

The DRB alleles in the first ten supertypes are gathered at the DRB1 locus. The DRB alleles in the ST51, ST52 and ST53 supertypes are gathered at the DRB5 , DRB3 and DRB4 locus, respectively.

4.2 Serotype designation of HLA-DRB alleles

There is a historically developed classification, based on extensive work of medical labs and organizations, called serotypes. This classification is oriented to immunology and diseases associated to gene variation in humans. It uses peptide binding data, 3D structure, X-ray diffraction and all tools available. When the confidence level is sufficiently high, WHO assigns a serotype to an allele as in Table 5 where a number prefixed by DR follows the name of that allele.

There are four expressed DRB genes (DRB1/DRB3/DRB4/DRB5) in the HLA-DRB region [16]. The DRB1 gene/locus is much more polymorphic than the DRB3/DRB4/DRB5 genes/loci [3]. More than 800 allelic variants are derived from the exon 2 of the DRB genes in humans [9]. The WHO Nomenclature Committee for Factors of the HLA System assigns an official name for each identified allele sequence, e.g. DRB1*01:01. The characters before the separator * describe the name of the gene, the first two digits correspond to the allele family and the third and fourth digits correspond to a specific HLA protein. See Table 5 for examples of how the alleles are named. If two HLA alleles belong to the same family, they often correspond to the same serological antigen, and thus the first two digits are meant to suggest serological types.

4.3 Comparison of identified supertypes to designated serotypes

In section 4.1, we have identified thirteen supertypes and in section 4.2 we have introduced the WHO assigned serotypes. In the following, we compare our classification results, the supertypes, with the WHO designated classification, the serotypes.

We have numbered our supertypes with prefix ”ST” parallel to the serotype numbering using the cluster tree. The detail information of DRB alleles and serological types for these 13 supertypes are given in Table 5. Our supertype clustering recovers the WHO serotype classification and provides further insight into the classification of DRB alleles which are not assigned serotypes. There are 559 DRB alleles in Table 5, and only 138 DRB alleles have WHO assigned serotypes. Table 6 gives the relationship between the broad serological type and the split serological type. As shown in Table 5 and Table 6, our supertypes assigned to these 138 DRB alleles are in exact agreement with the WHO assigned broad serological types. Extensive medical/biological information was used by WHO to assign serological type whereas solely DRB amino acid sequences was used in our supertype clustering. All alleles with WHO assigned DR52, DR3, DR6, DR8, DR4, DR2, DR5, DR53, DR9, DR7, DR51, DR10 and DR1-serotype are classified, respectively, into the ST52, ST3, ST6, ST8, ST4, ST2, ST5, ST53, ST9, ST7, ST51, ST10 and ST1-supertype. For the other 461 alleles in the bins, they are not assigned serotypes by WHO in the HLA dictionary 2008, however, WHO have suggested their serotypes according to their official names or allele families, that is if two DRB alleles are in the same family, they are suggested to be assigned the same serotype. And our clustering confirms that this suggestion is reasonable which can be checked from the bins.

We make some remarks on Figure 1 and Table 5 as follows.

ST52: This supertype consists of exactly the DRB3 alleles with the exception of DRB1*0338 (a new allele and unassigned by WHO [14]).

ST3: This supertype consists of cluster 2 and cluster 3 in the cluster tree and contains 63 DRB1*03 alleles with two exceptions of DRB3*0115 and DRB1*1525. The DRB3*0115 is grouped with the DRB1*03 alleles in a number of different experiments done by us, and the DRB1*1525 is a new allele and unassigned by WHO. Here, the DR3-serotype is a broad serotype which consists of three split serotypes, DR3, DR17 and DR18 (see Table 6).

ST6: This supertype consists of cluster 4 and cluster 5 and consists of exactly 103 DRB1*14 alleles. Here, the DR6-serotype is a broad serotype which consists of five split serotypes, DR6, DR13, DR14, DR1403 and DR1404.

ST8: This supertype consists of cluster 6 and cluster 7 and mainly contains 46 DRB1*08 alleles (The serological designation of DRB1*1415 is DR8 by WHO.). The unassigned alleles DRB1*1425, DRB1*1440, DRB1*1442, DRB1*1469, DRB1*1477 and DRB1*1484 are DRB1*14 alleles, but they are classified into the ST8 supertype. Both DRB1*14116 and DRB1*14102 are new allele sequences that do not exist in the tables of [14, 22] and they are classified into the ST8 supertype too.

Supertypes 52, 4, 2, 5, 53, 9, 7, 51, 10 and 1 correspond, respectively, to cluster 1, 8, 9, 10, 11, 12, 13, 14, 15 and 16 in the cluster tree.

ST4: This supertype consists of exactly 99 DRB1*04 alleles.

ST2: This supertype consists of 53 DRB1*15 alleles and 16 DRB1*16 alleles. Here, the DR2-serotype is a broad serotype which consists of three split serotypes, DR2, DR15 and DR16.

ST5: This supertype contains exactly 29 DRB1*12 alleles. The DRB1*0832 is undefined by experts in [14], but its serological designation by the neural network algorithm [21] is DR8 or DR12. We classify it into the ST5 supertype. The DR5-serotype is a broad serotype which consists of two split serotypes, DR11 and DR12.

ST53: This supertype consists of exactly the DRB4 alleles.

ST9: This supertype contains exactly the DRB1*09 alleles with the exception of DRB5*0112. The DRB5*0112 is undefined by experts in [14]. And from a number of different experiments done by us, DRB5*0112 is clustered with the DRB1*09 family of alleles.

ST7: This supertype consists of exactly 19 DRB1*07 alleles.

ST51: This supertype consists of exactly 15 DRB5 alleles.

ST10: This supertype is the smallest supertype and consists of exactly 3 DRB1*10 alleles.

ST1: This supertype consists of exactly 36 DRB1*01 alleles. Here, the DR1-serotype is a broad serotype which consists of two split serotypes, DR1 and DR103.

For the DRB alleles, there are thirteen broad serotypes given by WHO, and our clustering classifies all alleles which are assigned the same broad serotype to the same supertype. And for the alleles which are not assigned serotypes, our supertypes confirm the nomenclature of WHO.

As can be seen from Figure 1, the ST52 supertype is closest to the ST3 supertype. The ST53 supertype is closest to the ST9 and ST7 supertypes. The ST51 supertype is closest to the ST10 and ST1 supertypes.

4.4 Previous work in perspective

In 1999, Sette and Sidney asserted that all HLA I alleles can be classified into nine supertypes [34, 37]. This classification is defined based on the structural motifs derived from experimentally determine binding data. The alleles in the same supertype comprise the same peptide binding motifs and bind to largely overlapping set of peptides. Essentially, the supertype classification problem is to identify peptides that can bind to a group of HLA molecules. Besides many works on HLA class I supertype classification, some works have been proposed to identify supertypes for HLA class II. In 1998, through analyzing a large set of biochemical synthetic peptides and a panel of HLA-DR binding assays, Southwood et al. [39] asserted that seven common HLA-DR alleles, e.g. DRB1*0101, DRB1*0401, DRB1*0701, DRB1*0901, DRB1*1302, DRB1*1501 and DRB5*0101 had similar peptide binding specificity and should be grouped in one supertype. By the use of HLA ligands, Lund et al. [19] clustered 50 DRB alleles into nine supertypes by a Gibbs Sampling algorithm. Both of them used peptide binding data and this resulted in the limited number of DRB alleles available for classification. The work of Doytchinova and Flower [7], classified 347 DRB alleles into 5 supertypes by the use of both protein sequence and 3D structural data. Ou et al. [26]. defined seven supertypes based on the similarity of function rather than sequence or structure. To our knowledge, our study is the first to identify HLA-DR supertypes solely based on DRB amino acid sequence data.

Supertype	**Allele**	**Serotype**	**Allele**	**Serotype**	**Allele**	**Serotype**
ST52	**Cluster 1**					
	DRB3*0101(2)	DR52	DRB3*0108(U.)	-	DRB3*0212(U.)	-
	DRB3*0106(s.s.)	DR52	DRB3*0102(s.s.)	-	DRB3*0226	-
	DRB3*0110(s.s.)	DR52	DRB3*0112	-	DRB3*0222(U.)	-
	DRB3*0301	DR52	DRB3*0105(U.)	-	DRB3*0204(U.)	-
	DRB3*0209	DR52	DRB3*0103(s.s.)(U.)	-	DRB3*0213(U.)	-
	DRB3*0302(s.s.)	DR52	DRB3*0113	-	DRB3*0215(U.)	-
	DRB3*0107(s.s.)	DR52	DRB3*0111(U.)	-	DRB3*0218(U.)	-
	DRB3*0203(s.s.)	DR52	DRB3*0114	-	DRB3*0205(U.)	-
	DRB3*0211	DR52	DRB3*0303	-	DRB3*0225	-
	DRB3*0201(2)	DR52	DRB3*0109(U.)	-	DRB3*0219(U.)	-
	DRB3*0202(2)	DR52	DRB3*0206(s.s.)	-	DRB3*0216(U.)	-
	DRB3*0210	DR52	DRB3*0220(U.)	-	DRB3*0221(U.)	-
	DRB3*0208(s.s.)	DR52	DRB3*0223	-	DRB3*0227	-
	DRB3*0207(s.s.)	DR52	DRB3*0217(U.)	-		
	DRB1*0338	-	DRB3*0214(U.)	-		
ST3	**Cluster 2**					
	DRB1*0323	DR3	DRB1*0334	-	DRB1*0358	-
	DRB1*0301(2)	DR17	DRB1*0364	-	DRB1*0308	-
	DRB1*0305	DR3	DRB1*0361	-	DRB1*0326	-
	DRB1*0311	DR17	DRB1*0332	-	DRB1*0313	-
	DRB1*0304	DR17	DRB1*0328	-	DRB1*0360	-
	DRB1*0306	DR3	DRB1*0362	-	DRB1*0324	-
	DRB1*0307	DR3	DRB1*0346	-	DRB1*0352	-
	DRB1*0314	DR3	DRB1*0336	-	DRB1*0365	-
	DRB1*0315	DR3	DRB1*0357	-	DRB1*0329	-
	DRB1*0312(s.s.)	DR3	DRB1*0339	-	DRB1*0327	-
	DRB1*0302	DR18	DRB1*0333	-	DRB1*0353	-
	DRB1*0303	DR18	DRB1*0319	-	DRB1*0321	-
	DRB1*0310	DR17	DRB1*0348	-	DRB1*0343	-

	DRB1*0342	-	DRB1*0363	-	DRB1*0330	-
	DRB1*0345	-	DRB1*0322	-	DRB1*0325	-
	DRB1*0355	-	DRB1*0309	-	DRB1*0344	-
	DRB1*0359	-	DRB1*0337	-	DRB1*0331	-
	DRB1*0354	-	DRB1*0351	-	DRB1*0335	-
	DRB1*0320	-	DRB1*0347	-	DRB3*0115	-
	DRB1*0356	-	DRB1*0318	-	DRB1*0316(s.s.)	-
	Cluster 3					
	DRB1*1525	-	DRB1*0340	-	DRB1*0317	-
	DRB1*0349	-	DRB1*0341	-		
ST6	**Cluster 4**					
	DRB1*1410	DR14	DRB1*1482	-	DRB1*1472	-
	DRB1*1401(4)	DR14	DRB1*1462	-	DRB1*14101	-
	DRB1*1426	DR14	DRB1*1470	-	DRB1*1434	-
	DRB1*1407	DR14	DRB1*1438	-	DRB1*1423	-
	DRB1*1460	DR14	DRB1*14112	-	DRB1*1445	-
	DRB1*1450	DR14	DRB1*1490	-	DRB1*1443	-
	DRB1*1404	DR1404	DRB1*1486	-	DRB1*1456	-
	DRB1*1449	DR14	DRB1*1497	-	DRB1*14103	-
	DRB1*1411	DR14	DRB1*1435	-	DRB1*1444	-
	DRB1*1408	DR14	DRB1*1455	-	DRB1*1496	-
	DRB1*1414	DR14	DRB1*1431	-	DRB1*14100	-
	DRB1*1405	DR14	DRB1*1493	-	DRB1*1436	-
	DRB1*1420	DR14	DRB1*1428	-	DRB1*1465	-
	DRB1*1422	DR14	DRB1*1471	-	DRB1*1464	-
	DRB1*1416	DR6	DRB1*1468	-	DRB1*1495	-
	DRB1*1439	-	DRB1*1432	-	DRB1*1459	-
	DRB1*1499	-	DRB1*14111	-	DRB1*1491	-
	DRB1*1461	-	DRB1*14104	-	DRB1*1441	-
	DRB1*14117	-	DRB1*1458	-	DRB1*1437	-
	DRB1*1487	-	DRB1*1473	-	DRB1*1457	-
	DRB1*1475	-	DRB1*1479	-	DRB1*14105	-
	DRB1*1488	-	DRB1*14107	-	DRB1*1474	-
	DRB1*14110	-	DRB1*1476	-		
	Cluster 5					
	DRB1*1419	DR14	DRB1*1452	-	DRB1*1433	-
	DRB1*1402	DR14	DRB1*14108	-	DRB1*1424	-
	DRB1*1429	DR14	DRB1*1483	-	DRB1*14109	-
	DRB1*1406	DR14	DRB1*1481	-	DRB1*14115	-
	DRB1*1418	DR6	DRB1*1494	-	DRB1*1467	-
	DRB1*1413	DR14	DRB1*1447	-	DRB1*1498	-
	DRB1*1421	DR14	DRB1*1451	-	DRB1*1463	-
	DRB1*1417	DR6	DRB1*14106	-	DRB1*1485	-
	DRB1*1427	DR14	DRB1*1489	-	DRB1*1478	-
	DRB1*1403	DR1403	DRB1*1430	-	DRB1*1448	-
	DRB1*1412	DR14	DRB1*1409	-		
	DRB1*1446	-	DRB1*1480	-		
ST8	**Cluster 6**					
	DRB1*1442(U.)	-				
	Cluster 7					
	DRB1*0809	DR8	DRB1*1477	-	DRB1*0808	-
	DRB1*1415	DR8	DRB1*1440	-	DRB1*0844	-
	DRB1*0814	DR8	DRB1*1484	-	DRB1*0835	-
	DRB1*0812	DR8	DRB1*0846	-	DRB1*0836	-
	DRB1*0803	DR8	DRB1*0848	-	DRB1*0847	-
	DRB1*0810	DR8	DRB1*0819	-	DRB1*0825	-
	DRB1*0817	DR8	DRB1*0827	-	DRB1*0834	-
	DRB1*0811	DR8	DRB1*0829	-	DRB1*0828	-
	DRB1*0801	DR8	DRB1*0837	-	DRB1*0845	-

	DRB1*0807	DR8	DRB1*0839	-	DRB1*0830	-
	DRB1*0806	DR8	DRB1*0822	-	DRB1*0824	-
	DRB1*0805	DR8	DRB1*0815	-	DRB1*0820(U.)	-
	DRB1*0818	DR8	DRB1*0840	-	DRB1*14116	-
	DRB1*0816	DR8	DRB1*0838	-	DRB1*14102	-
	DRB1*0802	DR8	DRB1*0826	-	DRB1*0842	-
	DRB1*0804	DR8	DRB1*0843	-	DRB1*0841	-
	DRB1*0813	DR8	DRB1*0833	-	DRB1*1425	-
	DRB1*0821	-	DRB1*0823	-	DRB1*1469	-
ST4	**Cluster 8**					
	DRB1*0420(s.s.)	DR4	DRB1*0438	-	DRB1*0490	-
	DRB1*0401	DR4	DRB1*0434	-	DRB1*0487	-
	DRB1*0464	DR4	DRB1*0475	-	DRB1*0430	-
	DRB1*0408	DR4	DRB1*0476	-	DRB1*0448	-
	DRB1*0416	DR4	DRB1*0472	-	DRB1*0467	-
	DRB1*0426	DR4	DRB1*0435	-	DRB1*0483	-
	DRB1*0442	DR4	DRB1*0443	-	DRB1*0480	-
	DRB1*0432(s.s.)	DR4	DRB1*0479	-	DRB1*0462	-
	DRB1*0423	DR4	DRB1*0440	-	DRB1*0457	-
	DRB1*0404	DR4	DRB1*0470	-	DRB1*0497	-
	DRB1*0413	DR4	DRB1*0444	-	DRB1*0463	-
	DRB1*0431	DR4	DRB1*0456	-	DRB1*0498	-
	DRB1*0403	DR4	DRB1*0455	-	DRB1*0449	-
	DRB1*0407(2)	DR4	DRB1*0433	-	DRB1*04102	-
	DRB1*0429	DR4	DRB1*0439	-	DRB1*0441	-
	DRB1*0424	DR4	DRB1*0460	-	DRB1*0446	-
	DRB1*0409	DR4	DRB1*0450	-	DRB1*0485	-
	DRB1*0405	DR4	DRB1*0496	-	DRB1*0478	-
	DRB1*0410	DR4	DRB1*0451	-	DRB1*0465	-
	DRB1*0428	DR4	DRB1*0471	-	DRB1*0491	-
	DRB1*0417	DR4	DRB1*04100	-	DRB1*0468	-
	DRB1*0411	DR4	DRB1*0488	-	DRB1*0477	-
	DRB1*0422	DR4	DRB1*0493	-	DRB1*0484	-
	DRB1*0406	DR4	DRB1*0427	-	DRB1*0447	-
	DRB1*0421	DR4	DRB1*0452	-	DRB1*0436	-
	DRB1*0419	DR4	DRB1*04101	-	DRB1*0454	-
	DRB1*0425(s.s.)	DR4	DRB1*0474	-	DRB1*0437	-
	DRB1*0414	DR4	DRB1*0495	-	DRB1*0453	-
	DRB1*0402	DR4	DRB1*0459	-	DRB1*0418	-
	DRB1*0415	DR4	DRB1*0473	-	DRB1*0458	-
	DRB1*0499	-	DRB1*0461	-	DRB1*0486	-
	DRB1*0482	-	DRB1*0445	-	DRB1*0412	-
	DRB1*0466	-	DRB1*0489	-	DRB1*0469	-
ST2	**Cluster 9**					
	DRB1*1501(2)	DR15	DRB1*1533	-	DRB1*1548	-
	DRB1*1505	DR15	DRB1*1553	-	DRB1*1512	-
	DRB1*1506	DR15	DRB1*1524	-	DRB1*1515	-
	DRB1*1503	DR15	DRB1*1509	-	DRB1*1557	-
	DRB1*1508	DR2	DRB1*1549	-	DRB1*1511	-
	DRB1*1502(2)	DR15	DRB1*1541	-	DRB1*1538	-
	DRB1*1504	DR15	DRB1*1540	-	DRB1*1529	-
	DRB1*1507	DR15	DRB1*1523	-	DRB1*1545	-
	DRB1*1602	DR16	DRB1*1518	-	DRB1*1554	-
	DRB1*1605(s.s.)	DR16	DRB1*1537	-	DRB1*1510	-
	DRB1*1601	DR16	DRB1*1514	-	DRB1*1521	-
	DRB1*1609	DR16	DRB1*1544	-	DRB1*1612	-
	DRB1*1603	DR2	DRB1*1526	-	DRB1*1617	-
	DRB1*1604	DR16	DRB1*1539	-	DRB1*1611	-
	DRB1*1528	-	DRB1*1530	-	DRB1*1614	-

	DRB1*1535	-	DRB1*1531	-	DRB1*1618	-
	DRB1*1532	-	DRB1*1556	-	DRB1*1610	-
	DRB1*1542	-	DRB1*1555	-	DRB1*1608	-
	DRB1*1551	-	DRB1*1516	-	DRB1*1615	-
	DRB1*1552	-	DRB1*1522	-	DRB1*1607	-
	DRB1*1536	-	DRB1*1546	-	DRB1*1616	-
	DRB1*1520	-	DRB1*1547	-	DRB1*1527	-
	DRB1*1543	-	DRB1*1558	-	DRB1*1534	-
ST5	**Cluster 10**					
	DRB1*1202	DR12	DRB1*1215	-	DRB1*1230	-
	DRB1*1201(4)	DR12	DRB1*1219	-	DRB1*1207	-
	DRB1*1203	DR12	DRB1*1216	-	DRB1*1229	-
	DRB1*1205	DR12	DRB1*1221	-	DRB1*1234	-
	DRB1*1220	-	DRB1*1208	-	DRB1*1222	-
	DRB1*1233	-	DRB1*1212	-	DRB1*1223	-
	DRB1*1218	-	DRB1*1225	-	DRB1*1227	-
	DRB1*1213	-	DRB1*1211	-	DRB1*1209	-
	DRB1*1232	-	DRB1*1228	-	DRB1*1204	-
	DRB1*1226	-	DRB1*1214	-	DRB1*0832(U.)	-
ST53	**Cluster 11**					
	DRB4*0101(3)	DR53	DRB4*0104(U.)	-	DRB4*0107(U.)	-
	DRB4*0105(s.s.)	DR53	DRB4*0102(s.s.)(U.)	-	DRB4*0108	-
ST9	**Cluster 12**					
	DRB1*0901	DR9	DRB1*0912	-	DRB1*0915	-
	DRB1*0905	DR9	DRB1*0906	-	DRB1*0911	-
	DRB1*0910	-	DRB1*0908	-	DRB1*0914	-
	DRB1*0916	-	DRB1*0904	-	DRB5*0112(U.)	-
	DRB1*0907	-	DRB1*0903	-	DRB1*0902	-
	DRB1*0909	-	DRB1*0913	-		
ST7	**Cluster 13**					
	DRB1*0703	DR7	DRB1*0721	-	DRB1*0708	-
	DRB1*0701	DR7	DRB1*0716	-	DRB1*0711	-
	DRB1*0709	DR7	DRB1*0713	-	DRB1*0717	-
	DRB1*0704	DR7	DRB1*0714	-	DRB1*0707	-
	DRB1*0715	-	DRB1*0712	-	DRB1*0706	-
	DRB1*0719	-	DRB1*0720	-		
	DRB1*0705	-	DRB1*0718	-		
ST51	**Cluster 14**					
	DRB5*0101	DR51	DRB5*0104(U.)	-	DRB5*0106(U.)	-
	DRB5*0102	DR51	DRB5*0103(U.)	-	DRB5*0111(U.)	-
	DRB5*0107(s.s.)	DR51	DRB5*0113(U.)	-	DRB5*0204(U.)	-
	DRB5*0202	DR51	DRB5*0109(s.s.)(U.)	-	DRB5*0203(U.)	-
	DRB5*0105(U.)	-	DRB5*0114	-	DRB5*0205(U.)	-
ST10	**Cluster 15**					
	DRB1*1001	DR10	DRB1*1003	-	DRB1*1002	-
ST1	**Cluster 16**					
	DRB1*0107	DR1	DRB1*0120	-	DRB1*0135	-
	DRB1*0101	DR1	DRB1*0127	-	DRB1*0111	-
	DRB1*0102	DR1	DRB1*0112	-	DRB1*0117	-
	DRB1*0104	DR1	DRB1*0128	-	DRB1*0118	-
	DRB1*0109	DR1	DRB1*0136	-	DRB1*0115	-
	DRB1*0103	DR103	DRB1*0131	-	DRB1*0106	-
	DRB1*0113	DR1	DRB1*0132	-	DRB1*0126	-
	DRB1*0122	-	DRB1*0119	-	DRB1*0137	-
	DRB1*0124	-	DRB1*0130	-	DRB1*0123	-
	DRB1*0110	-	DRB1*0121	-	DRB1*0108	-
	DRB1*0129	-	DRB1*0105	-	DRB1*0114	-

DRB1*0134	-	DRB1*0125	-	DRB1*0116	-

Table 6: Bins of HLA-DR alleles with split serological types assigned by the World Health Organization.

The split serological types are obtained from [14]. The left column indicates the supertypes defined by the cluster tree. Remark on the labels for the alleles: WHO labeled split serological type is marked in square bracket; “(U.)” stands for “undefined” marked by the experts in [14]; “(s.s.)” indicates that the normal forms of the allele is shorter than 81 amino acids; “(n)” with $n = 1, 2, \cdots$ indicates that the normal form is shared by n alleles.

HLA-DRB1 serological families		
Broad Serotype	**Split serotype**	**Alleles**
DR1	DR1 DR103	DRB1*01 DRB1*0103
DR2	DR2 DR15 DR16	DRB1*1508, *1603 DRB1*15 DRB1*16
DR3	DR3 DR17 DR18	DRB1*0305, *0306, *0307, *0312, *0314, *0315, *0323 DRB1*0301, *0304, *0310, *0311 DRB1*0302, *0303
DR4	DR4	DRB1*04
DR5	DR11 DR12	DRB1*11 DRB1*12
DR6	DR6 DR13 DR14 DR1403 DR1404	DRB1*1416, *1417, *1418 DRB1*13, *1453 DRB1*14, *1354 DRB1*1403 DRB1*1404
DR7	DR7	DRB1*07
DR8	DR8	DRB1*08, *1415
DR9	DR9	DRB1*09
DR10	DR10	DRB1*10

DRB3/4/5 serological families	
Serotype	**Alleles**
DR51	DRB5*01,02
DR52	DRB3*01,02,03
DR53	DRB4*01

Table 7: Overview of the broad serological types in connection with the split serological types assigned by the World Health Organization. The serological type information listed in this table was extracted from the tables 4 and 5 given in [14]. This table summarizes the allele and serotype information given in the first and third columns of tables 4 and 5.

Acknowledgment

The authors would like to thank Yiming Cheng and Shuaicheng Li for helpful discussions. In particular, Shuaicheng Li pointed out to us that the portions of DRB alleles that contact with peptides can be obtain from the non-aligned DRB amino acid sequences by the use of two markers, ”RFL” and ”TVQ”. We thank Morten Nielsen for his criticism on over fitting.

Appendix

A The BLOSUM62-2 Matrix

We list the whole BLOSUM62-2 matrix in Table 8. Table 9 explains the amino acids denoted by the capital letters.

	A	R	N	D	C	Q	E	G	H	I
A	3.903	0.613	0.588	0.545	0.868	0.757	0.741	1.057	0.569	0.632
R	0.613	6.666	0.859	0.573	0.309	1.406	0.961	0.450	0.917	0.355
N	0.588	0.859	7.094	1.554	0.398	1.001	0.911	0.864	1.222	0.328
D	0.545	0.573	1.554	7.398	0.301	0.897	1.688	0.634	0.679	0.339
C	0.868	0.309	0.398	0.301	19.577	0.366	0.286	0.420	0.355	0.653
Q	0.757	1.406	1.001	0.897	0.366	6.244	1.902	0.539	1.168	0.383
E	0.741	0.961	0.911	1.688	0.286	1.902	5.470	0.481	0.960	0.331
G	1.057	0.450	0.864	0.634	0.420	0.539	0.481	6.876	0.493	0.275
H	0.569	0.917	1.222	0.679	0.355	1.168	0.960	0.493	13.506	0.326
I	0.632	0.355	0.328	0.339	0.653	0.383	0.331	0.275	0.326	3.998
L	0.602	0.474	0.310	0.287	0.642	0.477	0.373	0.285	0.381	1.694
K	0.775	2.077	0.940	0.784	0.349	1.554	1.308	0.589	0.779	0.396
M	0.723	0.623	0.475	0.346	0.611	0.864	0.500	0.395	0.584	1.478
F	0.465	0.381	0.354	0.299	0.439	0.334	0.331	0.341	0.652	0.946
P	0.754	0.482	0.500	0.599	0.380	0.641	0.679	0.477	0.473	0.385
S	1.472	0.767	1.232	0.914	0.738	0.966	0.950	0.904	0.737	0.443
T	0.984	0.678	0.984	0.695	0.741	0.791	0.741	0.579	0.558	0.780
W	0.417	0.395	0.278	0.232	0.450	0.509	0.374	0.422	0.444	0.409
Y	0.543	0.556	0.486	0.346	0.434	0.611	0.496	0.349	1.798	0.630
V	0.936	0.420	0.369	0.337	0.756	0.467	0.429	0.337	0.339	2.418

	L	K	M	F	P	S	T	W	Y	V
A	0.602	0.775	0.723	0.465	0.754	1.472	0.984	0.417	0.543	0.936
R	0.474	2.077	0.623	0.381	0.482	0.767	0.678	0.395	0.556	0.420
N	0.310	0.940	0.475	0.354	0.500	1.232	0.984	0.278	0.486	0.369
D	0.287	0.784	0.346	0.299	0.599	0.914	0.695	0.232	0.346	0.337
C	0.642	0.349	0.611	0.439	0.380	0.738	0.741	0.450	0.434	0.756
Q	0.477	1.554	0.864	0.334	0.641	0.966	0.791	0.509	0.611	0.467
E	0.373	1.308	0.500	0.331	0.679	0.950	0.741	0.374	0.496	0.429
G	0.285	0.589	0.395	0.341	0.477	0.904	0.579	0.422	0.349	0.337
H	0.381	0.779	0.584	0.652	0.473	0.737	0.558	0.444	1.798	0.339
I	1.694	0.396	1.478	0.946	0.385	0.443	0.780	0.409	0.630	2.418
L	3.797	0.428	1.994	1.155	0.371	0.429	0.660	0.568	0.692	1.314
K	0.428	4.764	0.625	0.344	0.704	0.932	0.793	0.359	0.532	0.457
M	1.994	0.625	6.481	1.004	0.424	0.599	0.794	0.610	0.708	1.269
F	1.155	0.344	1.004	8.129	0.287	0.440	0.482	1.374	2.769	0.745
P	0.371	0.704	0.424	0.287	12.838	0.756	0.689	0.282	0.364	0.443
S	0.429	0.932	0.599	0.440	0.756	3.843	1.614	0.385	0.558	0.565
T	0.660	0.793	0.794	0.482	0.689	1.614	4.832	0.431	0.573	0.981
W	0.568	0.359	0.610	1.374	0.282	0.385	0.431	38.108	2.110	0.374
Y	0.692	0.532	0.708	2.769	0.364	0.558	0.573	2.110	9.832	0.658
V	1.314	0.457	1.269	0.745	0.443	0.565	0.981	0.374	0.658	3.692

Table 8: The BLOSUM62-2 matrix.

A	Alanine	L	Leucine
R	Arginine	K	Lysine
N	Asparagine	M	Methionine
D	Aspartic acid	F	Phenylalanine
C	Cysteine	P	Proline
Q	Glutamine	S	Serine
E	Glutamic acid	T	Threonine
G	Glycine	W	Tryptophan
H	Histidine	Y	Tyrosine
I	Isoleucine	V	Valine

Table 9: The list of the amino acids.

From the Introduction, we see that the matrix Q can be recovered from the BLOSUM62-2 once the marginal probability vector p is available. The latter vector is obtained by

$$p = ([\text{BLOSUM62-2}])^{-1} v_1,$$

where $v_1 = (1, \cdots, 1) \in \mathbb{R}^{20}$ is a vector with all its coordinate being 1. The matrix Q can be obtained precisely from `http://www.ncbi.nlm.nih.gov/IEB/ToolBox/CPP_DOC/lxr/source/src/algo/blast/composition_adjustment/matrix_frequency_data.c#L391`.

References

[1] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.

[2] A. Baas, X.J. Gao, and G. Chelvanayagam. Peptide binding motifs and specificities for HLA-DQ molecules. *Immunogenetics*, 50:8–15, 1999.

[3] E.E. Bittar and N. Bittar, editors. *Principles of Medical Biology: Molecular and Cellular Pharmacology.* JAI Press Inc., 1997.

[4] F.A. Castelli, C. Buhot, A. Sanson, H. Zarour, S. Pouvelle-Moratille, C. Nonn, H. Gahery-Ségard, J.-G. Guillet, A. Ménez, B. Georges, and B. Maillère. HLA-DP4, the most frequent HLA II molecule, defines a new supertype of peptide-binding specificity. *J. Immunol.*, 169:6928–6934, 2002.

[5] F. Cucker and D.X. Zhou. *Learning Theory: An Approximation Theory Viewpoint.* Cambridge University Press, 2007.

[6] W.H.E. Day and H. Edelsbrunner. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of classification*, 1(1):7–24, 1984.

[7] I.A. Doytchinova and D.R. Flower. In silico identification of supertypes for class II MHCs. *J. Immunol.*, 174(11):7085–7095, 2005.

[8] Y. El-Manzalawy, D. Dobbs, and V. Honavar. On evaluating MHC-II binding peptide prediction methods. *PLoS One*, 3:e3268, 2008.

[9] M. Galan, E. Guivier, G. Caraux, N. Charbonnel, and J.-F. Cosson. A 454 multiplex sequencing method for rapid and reliable genotyping of highly polymorphic genes in large-scale studies. *BMC Genomics*, 11(296), 2010.

[10] Dan Graur and Wen-Hsiung Li. *Fundamentals of molecular evolution.* Sunderland, Mass.: Sinauer Associates, 2000.

[11] W.W. Grody, R.M. Nakamura, F.L. Kiechle, and C. Strom. *Molecular Diagnostics: Techniques and Applications for the Clinical Laboratory.* Academic Press, 2010.

[12] D. Haussler. Convolution kernels on discrete structures. Technical report, 1999.

[13] S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89:10915–10919, 1992.

[14] R. Holdsworth, C.K. Hurley, S.G. Marsh, M. Lau, H.J. Noreen, J.H. Kempenich, M. Setterholm, and M. Maiers. The HLA dictionary 2008: a summary of HLA-A, -B, -C, -DRB1/3/4/5, and -DQB1 alleles and their association with serologically defined HLA-A, -B, -C, -DR, and -DQ antigens. *Tissue Antigens*, 73(2):95–170, 2009.

[15] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, 1994.

[16] C.A. Janeway, P. Travers, M. Walport, and M.J. Shlomchik. *Immunobiology (5th Edition).* Garland Science, 2001.

[17] C. Leslie, E. Eskin, and W.S. Noble. The spectrum kernel: a string kernel for SVM protein classification. *Pacific Symposium on Biocomputing*, 7:566–575, 2002.

[18] H. H. Lin, G. L. Zhang, S. Tongchusak, E. L. Reinherz, and V. Brusic. Evaluation of MHC-II peptide binding prediction servers: applications for vaccine research. *BMC Bioinformatics*, 9 (Suppl 12):S22, 2008.

[19] O. Lund, M. Nielsen, C. Kesmir, A.G. Petersen, C. Lundegaard, P. Worning, C. Sylvester-Hvid, K. Lamberth, G. Røder, S. Justesen, S. Buus, and S. Brunak. Definition of supertypes for HLA molecules using clustering of specificity matrices. *Immunogenetics*, 55(12):797–810, 2004.

[20] O. Lund, M. Nielsen, C. Lundegaard, C. Keşmir, and S. Brunak. *Immunological Bioinformatics.* The MIT Press, 2005.

[21] M. Maiers, G.M. Schreuder, M. Lau, S.G. Marsh, M. Fernandes-Vi na, H. Noreen, M. Setterholm, and C. Katovich Hurley. Use of a neural network to assign serologic specificities to HLA-A, -B and -DRB1 allelic products. *Tissue Antigens*, 62(1):21–47, 2003.

[22] S.G.E. Marsh, E.D. Albert, W.F. Bodmer, R.E. Bontrop, B. Dupont, H.A. Erlich, M. Fernández-Vi na, D.E. Geraghty, R. Holdsworth, C.K. Hurley, M. Lau, K.W. Lee, B. Mach, M. Maiersj, W.R. Mayr, C.R. Müller, P. Parham,

E.W. Petersdorf, T. SasaZuki, J.L. Strominger, A. Svejgaard, P.I. Terasaki, J.M. Tiercy, and J. Trowsdale. Nomenclature for factors of the HLA system, 2010. *Tissue Antigens*, 75(4):291–455, 2010.

[23] M. Nielsen, S. Justesen, O. Lund, C. Lundegaard, and S. Buus. NetMHCIIpan-2.0: Improved pan-specific HLA-DR predictions using a novel concurrent alignment and weight optimization training procedure. *Immunome Research*, 6(1):9, 2010.

[24] M. Nielsen and O. Lund. NN-align. An artificial neural network-based alignment algorithm for MHC class II peptide binding prediction. *BMC Bioinformatics*, 10:296, 2009.

[25] M. Nielsen, C. Lundegaard, T. Blicher, B. Peters, A. Sette, S. Justesen, S. Buus, and O. Lund. Quantitative predictions of peptide binding to any HLA-DR molecule of known sequence: NetMHCIIpan. *PLoS Comput. Biol.*, 4(7):e1000107, 2008.

[26] D. Ou, L.A. Mitchell, and A.J. Tingle. A new categorization of HLA DR alleles on a functional basis. *Hum. Immunol.*, 59(10):665–676, 1998.

[27] J. Robinson, M.J. Waller, P. Parham, N. de Groot, R. Bontrop, L.J. Kennedy, P. Stoehr, and S.G. Marsh. IMGT/HLA and IMGT/MHC: Sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Res.*, 31(1):311–314, 2003.

[28] R. Sadiq and S. Tesfamariam. Probability density functions based weights for ordered weighted averaging (OWA) operators: An example of water quality indices. *European Journal of Operational Research*, 182(3):1350–1368, 2007.

[29] H. Saigo, J.-P. Vert, N. Ueda, and T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, jul 2004.

[30] H. Saigo, J.P. Vert, and T. Akutsu. Optimizing amino acid substitution matrices with a local alignment kernel. *BMC Bioinformatics*, 7:246, 2006.

[31] J. Salomon and D.R. Flower. Predicting class II MHC-peptide binding: a kernel based approach using similarity scores. *BMC Bioinformatics*, 7:501, 2006.

[32] B. Schölkopf and A. J. Smola. *Learning with Kernels.* The MIT Press, 2001.

[33] A. Sette, L. Adorini, S.M. Colon, S. Buus, and H.M. Grey. Capacity of intact proteins to bind to MHC class II molecules. *J Immunol.*, 143(4):1265–1267, 1989.

[34] A. Sette and J. Sidney. Nine major HLA class I supertypes account for the vast preponderance of HLA-A and -B polymorphism. *Immunogenetics*, 50(3-4):201–212, 1999.

[35] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis.* Cambridge University Press, 2004.

[36] J. Sidney, H.M. Grey, R.T. Kubo, and A. Sette. Practical, biochemical and evolutionary implications of the discovery of HLA class I supermotifs. *Immunol. Today*, 17(6):261–266, 1996.

[37] J. Sidney, B. Peters, N. Frahm, C. Brander, and A. Sette. HLA class I supertypes: a revised and updated classification. *BMC Immunology*, 9(1), 2008.

[38] S. Smale, L. Rosasco, J. Bouvrie, A. Caponnetto, and T. Poggio. Mathematics of the neural response. *Foundations of Computational Mathematics*, 10(1):67–91, 2010.

[39] S. Southwood, J. Sidney, A. Kondo, M.F. del Guercio, E. Appella, S. Hoffman, R.T. Kubo, R.W. Chesnut, H.M. Grey, and A. Sette. Several common HLA-DR types share largely overlapping peptide binding repertoires. *The Journal of Immunology*, 160(7):3363–3373, 1998.

[40] Glenys Thomson, Nishanth Marthandan, Jill A. Hollenbach, Steven J. Mack, Henry A. Erlich, Richard M. Single, Matthew J. Waller, Steven G. E. Marsh, Paula A. Guidry, David R. Karp, Richard H. Scheuermann, Susan D. Thompson, David N. Glass, and Wolfgang Helmberg. Sequence feature variant type (SFVT) analysis of the HLA genetic association in juvenile idiopathic arthritis. In *Pacific Symposium on Biocomputing'2010*, pages 359–370, 2010.

[41] P. Wang, J. Sidney, C. Dow, B. Mothé, A. Sette, and B. Peters. A systematic assessment of MHC Class II peptide binding predictions and evaluation of a consensus approach. *PLoS Computational Biology*, 4:e1000048, 2008.

[42] R.R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Trans. on Systems, Man and Cybernetics*, 18(1):183–190, 1988.

[43] J.W. Yewdell and J.R. Bennink. Immunodominance in major histocompatibility complex class I-restricted T lymphocyte responses. *Annu Rev Immunol.*, 17:51–88, 1999.