Ligand classification on the example of HLA subtype specific binding epitope prediction

Dissertation zur Erlangung des akademischen Grades des Doktors der Naturwissenschaften (Dr. rer. nat.) eingereicht im Fachbereich Biologie, Chemie, Pharmazie der Freien Universität Berlin

eingereicht von
Henning Riedesel, aus Osnabrück
April 2009

Die Arbeit wurde unter Leitung von Prof. Dr. E.W. Knapp am Institut für Chemie und Biochemie vom Nov. 2002 bis zum Okt. 2008 angefertigt.

1. Gutachter : Prof. Dr. E.W. Knapp, Freie Universität Berlin

2. Gutachter : Prof. Dr. J. Selbig, Universität Potsdam

Tag der Disputation : 19.06.2009

Abstract

In the present work algorithms are introduced to classify the ability of peptides to bind or not to bind. We classify ligand peptides of human leukocyte antigen (HLA) -A0201 by a least square optimization method (LSM), which is based on Fisher's linear discriminant. The LSM is capable using either sequence based features or feature vectors of physico-chemical derived quantities or other numerical descriptors like pharmacophore fingerprints [1]. In this work sequence based features and physico-chemical derived features are applied. For the evaluation of HLA-A0201 binding peptides known binding peptides are extracted from public ligand databases [2, 3]. Non-binding peptides are generated of protein sequences with a randomized approach and a counter check with known binding peptides. For learning and prediction of HLA-A0201 binding peptides sequence based feature vectors are used. The LSM performs well for recognition and prediction of A0201 binding peptides compared to well established methods like support vector machine (SVM) [4]. Due to regularization terms the LSM performs good even in situations where learning data sets are small compared to the size of the available parameters or very asymmetric learning sets regarding the composition between binding and non-binding peptides. In the case of asymmetric learning sets it even can outperform the SVM.
For a in depth understanding of the binding of peptides to HLA-A0201 several crystal structures of this allele together with different ligand peptides bound have been examined in this work. Furthermore the results have been compared to crystal structures containing besides the HLA-A0201 and the ligand peptide a T-Cell receptor (TCR) attached. For allele A0201 it could be found that N- and C- terminal residues of the bound peptide are closely attached to the HLA protein, while the central part is more flexibel to move in the HLA binding groove. The central part of the ligand peptide interacts predominantly with an attached TCR.
The Comparative Evaluation of Prediction Algorithms 2006 (CoEPrA) [5] is a competition for classification in machine learning. In four classification tasks learning and prediction sets of peptides were provided, which can be used for a binding prediction. Each data set provides sequence and physico-chemical features such that both type of descriptors could be used for the LSM. With sequence based features and a feature reduction based on principle component analysis (PCA) rankings in the top or middle field of the competition have been reached using the LSM. The application of the physico-chemical features required a strategy of feature reduction or selection since 643 physico-chemical features are provided per residue position. For feature reduction a lambda regularization term or the PCA was used. Results for the four different CoEPrA tasks achieved by the application of feature reduction and physico-chemical features are similar to the results obtained from the usage of sequence based features.
In the last part of this work a genetic algorithm (GA) is used for feature selection on the example of the four CoEPrA tasks. Small sets of features are selected to perform learning and prediction based on the LSM approach. The GA is generating a number of these feature sets called individuals. At the end of a GA run an enrichment of good performing feature sets can be observed. The difficulty to discriminate between good performing individuals and individuals, which show

learning by heart could not be solved reliably with different approaches tested in this work. Best performing individuals, which are generated for all four CoEPrA tasks are capable to reach a top ranking in the competition but it was not possible to identify those individuals with a satisfying accuracy.

Zusammenfassung

In der vorliegenden Arbeit werden Algorithmen vorgestellt, die zur Bindungsklassifizierung von Peptiden als mögliche Liganden benutzt werden. Die Differenzierung zwischen bindenden und nichtbindenden Peptiden ist in der Biochemie und der medizinischen Chemie von besonderer Bedeutung. In dieser Arbeit werden immun aktive Peptide (Antigene) untersucht, die an das Human Leukocyte Antigen (HLA) des Typs A-0201 binden. Dieser Komplex ist Teil des menschlichen Immunsystems und dient zur Erkennung und Abwehr von fremden Proteinen in körpereigenen Zellen. Die Klassifizierung der Peptide erfolgt durch die Methode der kleinsten Quadrate (LSM), die auf dem Verfahren der linearen Discrimianzanalyse nach Fisher basiert. Die Methode trainiert bekannte bindende und nichtbindende Peptide, um charakteristische Muster zu lernen und zu verallgemeinern. Die Peptide werden durch Merkmale, sogenannte Feature zu Deskriptoren zusammengefasst, welche mit den Vorhersageklassen (hier bindend und nichtbindend) korreliert werden sollen. Als Feature können sequenzbasierte Feature, physiko-chemische Feature oder andere numerische Feature wie "pharmacophore Fingerprints" benutzt werden. Im ersten Teil der Arbeit werden Sequenzen bekannter HLA-A0201 bindender Peptide aus öffentlich zugänglichen Datenbanken extrahiert. Nichtbindende Sequenzen werden zufällig aus der Sequenz von Proteinen generiert und um bekannte bindende Sequenzen bereinigt. Die erreichte Vorhersagegenauigkeit wird mit der alternativen Methode "Support Vector Machine" (SVM) verglichen. Im direkten Vergleich erreicht die LSM Methode eine ähnlich gute Wiedererkennungs- und Vorhersagegenauigkeit wie die SVM. Der Regularisierungsparameter λ_w verhindert das Auswendiglernen von kleinen Lerndatensätzen, wenn die Zahl der verwendeten featurebasierten Parameter hoch ist. Bei asymmetrischen Lerndatensätzen, wo sehr viel mehr Daten einer Klasse vorhanden sind, kann die LSM dank des Gewichtungsfaktors w^+ die SVM ausspielen. Vorhersage- und Wiedererkennungsrate sind in diesem Fall bei der LSM besser, können bei der SVM jedoch durch eine manuelle Korrektur ausgeglichen werden.
Im zweiten Teil der Arbeit werden Kristallstrukturen von HLA-A0201 mit gebundenen Peptiden unterschiedlicher Sequenz untersucht und verglichen mit Kristallstrukturen von HLA-A0201 die neben dem Peptidliganden einen gebundenen T-Zellrezeptor (TCR) enthalten. Für die Bindung des Peptids an das HLA-A0201 spielen Wechselwirkungen mit den N- und C- terminalen Peptidresiduen eine große Rolle. Die zentralen Residuen des Peptids weisen eine erhöhte Flexibilität in der Bindungstasche des A0201 Proteins auf. Dieser zentrale Bereich um die Positionen 4, 5 und 6 wechselwirkt mit einem vorhandenen TCR. Dabei werden Wechselwirkungen zwischen TCR und den Seitenkettenatomen des Peptids eingegangen.
Im letzten Teil der Arbeit werden Datensätze aus dem Bindungsvorhersagewettbewerb CoEPrA 2006 mit der LSM Methode untersucht. CoEPrA 2006 stellt in vier Klassifikationsaufgaben Lern-

und Vorhersagedatensätze von Peptiden mit Sequenzdaten und jeweils 643 physiko-chemische Feature pro Sequenzposition zur Verfügung. Anhand von sequenzbasierten Features und einer Featurereduktion mittels λ_w-Regularisierung oder Principle Component Analysis (PCA) werden Vorhersageergebnisse erzielt, die für eine Platzierung im Mittelfeld bis oberen Drittel des Teilnehmerfeldes von CoEPrA2006 reichen. Ähnliche Ergebnisse lassen sich mit den physiko-chemischen Features und einer Featurereduktion erreichen. Mithilfe eines genetischen Algorithmus (GA) wird Featureselektion anhand der physiko- chemischen Feature der CoEPrA Datensätze betrieben, um eine gezielte Reduktion des Parameterraumes zu erreichen. Dabei werden kleine Sätze von Features ausgewählt, die für die LSM zum Lernen und zur Vorhersage verwendet werden. Der GA erzeugt eine Reihe solcher Featuresätze, die Individuen genannt werden. Am Ende des GA wird eine Anreicherung von Individuen mit hoher bis guter Vorhersagegegenauigkeit erzielt. Als Schwierigkeit stellt sich jedoch die Unterscheidung von Individuen mit guter Vorhersagegenauigkeit und solchen Individuen, die auswendig lernen heraus, da Letztere in der Wiedererkennung und damit in der Testvorhersage gute bis sehr gute Ergebnisse erzielen. Die besten Individuen, die in den vier CoEPrA Aufgaben mit dem GA erzeugt werden sind gut genug, um ein Topplatzierung in der Bestenliste der Teilnehmer zu erreichen. Jedoch erlaubt keines der getesteten Verfahren in allen vier Aufgaben eine eindeutige Identifizierung der guten bis sehr guten Individuen aus der letzten Generation des GA.

Contents

List of Tables

List of Figures

Abbreviations

aa	amino acid
aL	(average) test Learning set
aT	(average) Test prediction set
AUROC	Area Under ROC Curve
CoEPrA	Comparative Evaluation of Prediction Algorithms
ERAP I	Endoplasmatic Reticulum AminoPeptidase I
$\boldsymbol{F^0}$	neutral group of features (balanced recognition)
$\boldsymbol{F^+}$	positive group of features (pref. binders recognition)
$\boldsymbol{F^-}$	negative group of features (pref. non-binders recognition)
FN	False Negatives
FP	False Positives
GA	Genetic Algorithm
HMM	Hidden Markov Models
HLA	Human Leukocyte Antigen (human type of MHC)
IBS	Independent Binding of Side-chains
L+	binding peptides from the learning set
L-	non-binding peptides from the learning set
LSM	Least Square (optimization) Method
LMCC	MCC value obtained for the learning set
MCC	Matthews Correlation Coefficient
MHC	Major Histocompatibility Complex
min	minimum value
P+	binding peptides from the prediction set
P-	non-binding peptides from the prediction set

P*	all peptides from the prediction set
PCA	Principle Component Analysis
PDB	Protein DataBase
pMHC	Major Histocompatibility Complex with ligand peptide bound
QSM	least Square Method with (additional) Quadratic terms
RMS	Root Mean Square
RMSD	Root Mean Square Deviation
ROC	Receiver Operating Characteristic
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TCR	T-Cell Receptor
tL	true Learning set
TN	True Negatives
TP	True Positives
tP	true Prediction set
var	variance of values

Chapter 1

Introduction

1.1 Protein Ligand binding

One major challenge in modern life sciences is to understand how protein complexes interact with ligands that trigger different kinds of responses on the molecular level inside of living cells. The main focus of interest is the receptor ligand binding, which is part of numerous molecular mechanisms i.e. the opening of ion channels, the initialization of the biosynthesis of transmitter molecules or the triggering of the immune response in mammals. Common to all those mechanisms is the docking of the ligand molecule into the binding pocket of the target compound, the receptor. On the atomic level the binding of two molecules is based on formation of numerous hydrogen bonds, electrostatic interactions and *van der Waals* interactions of hydrophobic regions. It is known that in some cases after the binding event the receptor molecule undergoes conformational changes which finally allow the actual mechanism to proceed. This process is called induced fit.
Modern medicine tries to mimic known ligands to receptors to induce certain physiological effects. Some drugs bind for instance to neuronal receptors in the brain and stimulate the emission of neurotransmitters. The demand of intermolecular interactions can go into both directions, to stimulate the physiological response by drug analoga (agonists) or to inhibit the stimulation by blocking the receptor and prevent the binding of cellular agonists (antagonist). Thus the understanding and design of binding ligands to a specific receptor protein is important for the development of new drugs. As there are usually millions or more possible candidates of ligands to find ligands that bind efficiently is evidently a difficult task.

1.1.1 In silico search for ligands

Since many years scientists use a computational approach to search for effective ligands in order to minimize time and costs of drug screening in the laboratory. Several methods were developed using different strategies.

Docking simulations use physical descriptions on the molecular level trying to fit a compound into the (supposed) active region of the receptor protein. A score based on the energy of the system, mainly Gibbs free energy ΔG, is calculated and has to be minimized. If available crystal structures of the ligand bound receptor molecules are used as starting conformation for the model of the active site of the receptor. The conformations of ligand and receptor molecule play an important role in finding optimal fits. New ligands can be derived from partially modified known binders. Besides the difficulty of identifying the binding pocket of the receptor and the optimal orientation of the ligand within this pocket, the flexibility of ligand and receptor raises

the need to investigate hundreds to thousands of different conformations of the molecules.

Indirect drug design is a similar approach to the direct drug design method of docking simulations, but without the explicit and implicit use of information of the receptor molecule. The idea is to use a number of known agonists and/or antagonists to find structural similarity of those molecules. Any potential ligand will be compared in terms of structural similarity to known ligands and scored. This method therefore requires knowledge about the 3D structure of a number of ligands known to interact with the receptor. If several binding modes of the ligand to the receptor are possible, a clustering of different ligands with respect to their favored binding modes has to be performed. As indirect drug design is not taking the binding pocket of the receptor into consideration, but just compares ligands, no information about the docking geometry is used.

For classification of molecules into binding or non binding classes **knowledge based methods** use descriptors of the ligands such as sequence information (if the ligands are peptides), physicochemical properties or pharmacophore fingerprints. It is essential for this methods to have a large number of known binding and non-binding ligands. In general this set of known ligands is partially used to train (learn) the method, while the other part is used to control the learning progress. Many different approaches are used for learning such as neuronal nets, hidden markov models, support vector machines, decision trees and other scoring functions.

1.2 Immune response in mammals

The main task of the immune system is to recognize infected or pathological modified cells and to destroy them. The adaptive immune system in mammals use the detection of foreign peptide pattern recognized by lymphocytes.

1.2.1 Immune response: Major Histocompatibility Complex

The *Major Histocompatibility Complex* (MHC) is a class of membrane proteins of the immune system in mammals. They play an important role in the detection of alien proteins inside of cells. Two classes of MHC have to be differentiated. MHC class I is present in almost every type of cell of the organism while class II is expressed on antigen presenting cells only. Class-I type MHCs are highly diverse and can be divided into several types and subtypes. In contrast to MHC class II proteins, MHC class I types are binding shorter peptide fragments of a more conserved length. One well known MHC class-I representative is the *Human Leukocyte Antigen* (HLA) subtype A0201 which is part of the HLA*02 main type. Every human individual has three different gene loci from each parent to express different MHC class I proteins. With respect to the gene loci, the MHC class I proteins are called HLA -A,-B or -C. As the loci are very polymorphic, for every loci a number of allels are known. Allels are referred in the following as types or subtypes. Every subtype has a different specificity against *antigens*, which are ligand peptides from the host cell.

The antigens are proteolytic degraded endogenous protein fragments, which characterize the protein pool present in the host cell like a fingerprint. Proteosomes cleave tagged proteins to form precursor peptides in the cytosol. Those 9-15 residue long peptides will be transported into the endoplasmatic reticulum via TAP transporter where the enzyme *endoplasmatic reticulum aminopeptidase I* (ERAP I) trims single residues from the N-terminal end of the precursor peptides [6]. If a peptide length of 8-9 residues is reached those peptides can be “loaded” into the

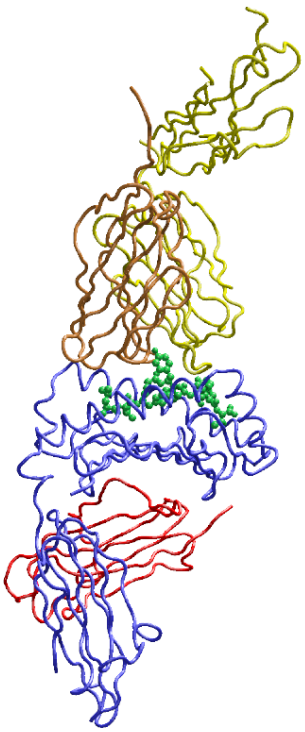


Figure 1.1: MHC-peptide-TCR crystal structure from 1AO7. Red and blue chains belong to HLA-0201 α and β chain, yellow and orange chains are part of the TCR. The bound peptide is colored in green.

MHC molecule. Finally the MHC-peptide complex is moving to the outer cell surface where it presents the ligand to *T-cell receptor* proteins (TCR), part of cytotoxic T-cells, which are screening the cell surface for those complexes. Each cell presents thousands of MHC-peptide complexes, which interact with the TCR. During this interaction the peptide ligand is temporary buried by both proteins (as shown in figure 1.1). Specific antigens presented to the TCR are stimulating the secretion of cytokins [7][8][9]. Since each T-cell screens many MHC-peptide complexes at the same time a multiple positive stimulation can trigger the immune response and lead to cytolysis of infected cells. Similar to the diversity of the MHC types in the human organism, the TCR is highly diverse as it is randomly recombined of TCR-$\alpha\beta$ genetic segments [10]. Ripening and selection of the TCR types occurs in the thymus [11].

1.3 Machine learning and classification

The terminology of machine learning covers many different methods such as neuronal networks, hidden Markov models (stochastic models), support vector machines, linear scoring functions and many more. All those methods have in common that description values have to be correlated with expectation / observation values. A learning data set is used to train the algorithm before it will be able to generalize correlation rules and to predict untrained data. To predict untrained data the trained system has to be provided with the description data of the prediction set.
A typical learning task is the classification of data. Other than the regression of data where i.e.

a binding constant is correlated to description values, the classification associates the description values to a limited number of observation states called classes. One example for a classification is the prediction of secondary structures in proteins. Typical structure motifs to discriminate are helix-, strand- and coil structures. The difficulty to compute multi-class predictions can be solved by defining three classes - helix, strand and coil - and treat the problem as three 2-class predictions. An example of a two-class classification problem is the binding prediction between ligands and receptor molecules. With a regression approach the prediction results can be understood as binding constants or binding probabilities of the ligand-receptor complex, while the classification approach discriminates between binding class (+1) or non-binding class (-1) ligands only.
For knowledge based classification it is important that the results to be expected can essentially depend on the quality and the completeness of the input training data [12]. The same is true for the features used to describe the data. When the features are compared to the vocabulary of a language, one would fail to gather information, which is necessary for classification, if the language used is not expressive enough. This is crucial, when it comes to feature selection with the genetic algorithm (see section 2.3.11), but it is also valid for any kind of classification algorithm.

1.3.1 Binding prediction of MHC antigens

This Ph.D. thesis focus on the binding prediction of ligands to MHC class I complex and in particular using subtype HLA-A0201 peptides. The method for prediction, which is presented in this work, was expanded to a more general level such that it is easily possible to apply it for any kind of drug/ligand-receptor binding classification. Although the main focus here are peptide ligands, any other synthetic ligands can be used as long as describing features can be derived for each ligand.
In the beginning of the studies for this Ph. D. work neuronal networks were used to classify binding ligands. The training mode of this method is time consuming and this heuristic approach is not computing exact solutions. Thus another approach using a linear scoring function based on Fisher's linear discriminant [13] was favored. The scoring function is derived by solving an optimized linear equation system defined by a symmetric "m by m" weight matrix. Each weight represents the occurrence of a certain feature pairing in the learning set. The expectation value defining the class of the molecule is correlated to this feature pattern. Historically simple hand optimized coefficient weight matrices have been introduced after performing binding studies on MHC-peptide complexes [14] [15] [2] trying to assign certain weights for a given amino acid type on a specific residue position of the bound peptide. This proceedure was supported by the commonly accepted IBS (*Independent Binding of Side-chains*) hypothesis [15][16], which assumes that neighboring residues have mainly no influence on the amount of binding strength accommodated by a single residue. IBS postulates that each residue position can be considered independent from neighboring residues, since almost no sidechain - sidechain interactions of ligand peptide residues occur.
To collect binding antigen peptides of HLA subtypes, publicly available MHC ligand (epitope) databases like SYFPEITHI [2] and MHCPEP [3] were used for this work. For non-binding peptides random sequences were generated implicitly filtering out those sequences listed in one of the specified databases of binders. The right distribution between binding and non-binding learning data is determining the quality of the final prediction since imbalanced learning sets can cause problems [17] in generalization. Therefore a weighting factor was introduced (see sec. 2.3.2d).

Alternatively binding constants determined by experimental binding studies can be used to separate ligands into classes of binding or non-binding peptides. For classifying peptides in this way, one has to define a threshold binding constant separating good from bad binders. This classification approach was used for the CoEPrA competition [5].

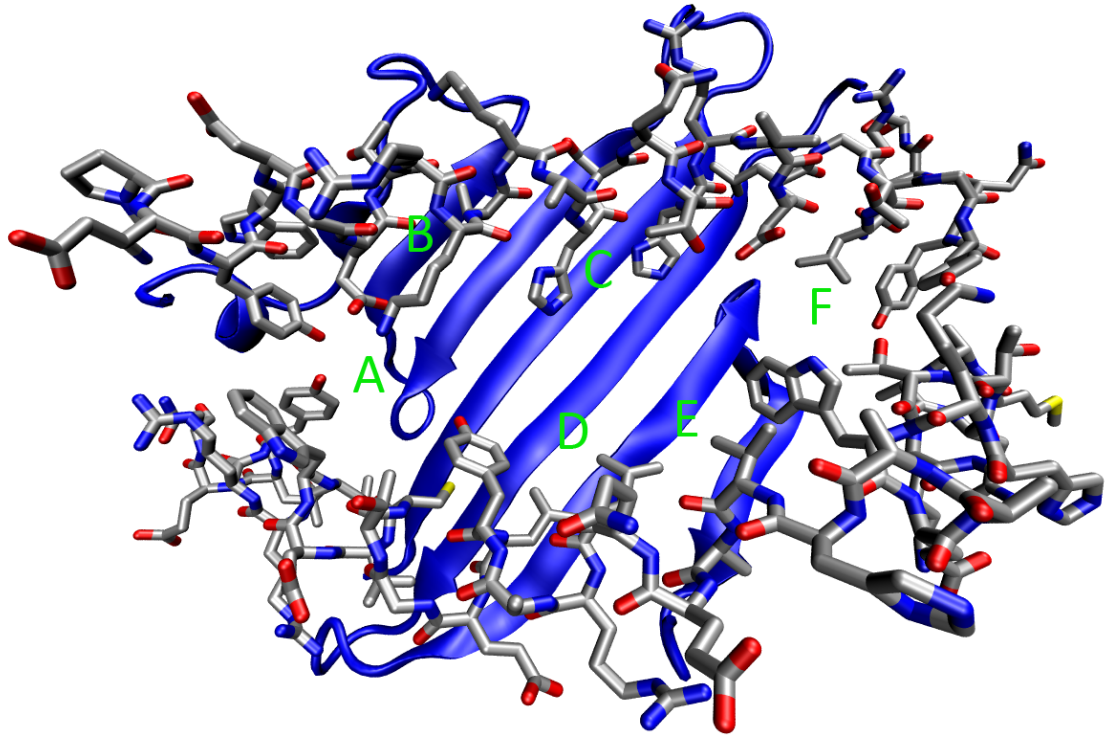


Figure 1.2: Anchor positions for ligand peptides in the HLA-A0201 binding pocket as suggested by Saper et. al. [18]. See also section 2.1.2 on page 8

Chapter 2

Material and Methods

2.1 Basics

2.1.1 Protein-Peptide interactions

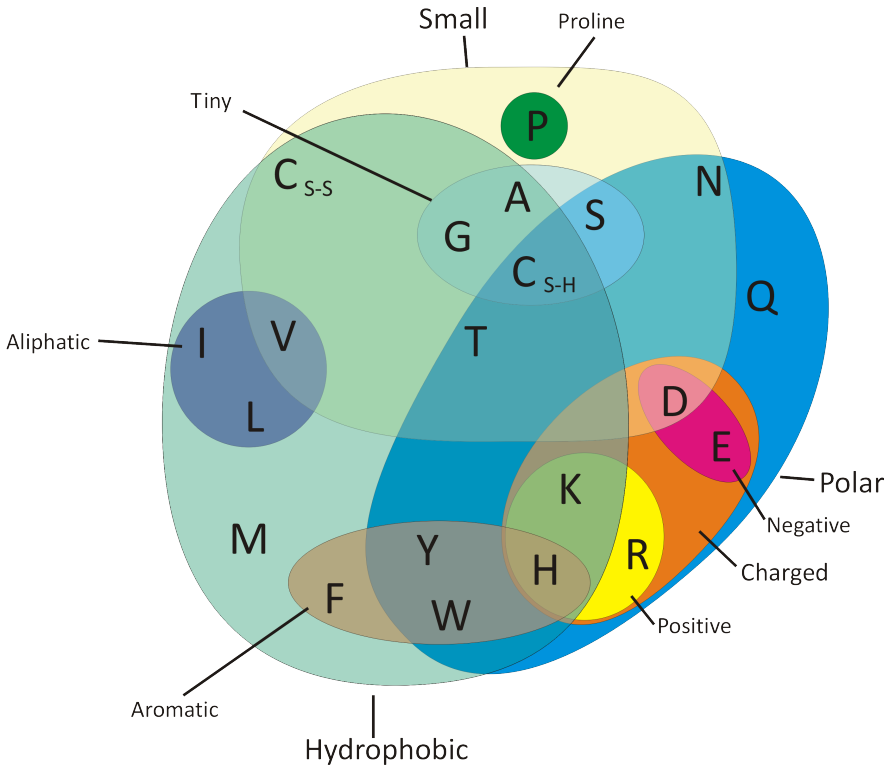


Figure 2.1: Characteristic properties of amino acids

Proteins and peptides are assembled out of smaller building blocks, the amino acids (aa). The amino acids are linked through amide bonds, which are formed in between the α- carboxyl and α-amino moiety of two amino acids. The amide bonds, also known as peptide bonds, in nature never occur between side chain and backbone atoms. Thus, the order of amino acids linked in a protein or peptide chain is always sequential and not branched. The basis set of nature contains 20 different amino acids, all of them differing in their side chains providing them with different

physico-chemical properties. Large hydrophobic side chains of amino acids are more likely buried in a protein than exposed to an aqueous surface while hydrophilic side chains favor the contact with water or other polar environments. Atoms of side chains of amino acids interact with each other as described by the Lennard Jones potential where uncharged atoms interact via long range attractive and short range repulsive forces. Charged or partially charged atoms interact via electrostatic interactions described by Coulomb's law. Polar atoms can i.e. form hydrogen bonds if a donor atom with covalently bounded hydrogen interacts with a hydrogen acceptor group. Hydrogen bond donors are amid-, hydroxy- or sulfuryl-groups at a certain distance to an acceptor nitrogen or oxygen atom. Salt bridges are formed by residues with opposite charges (acidic and base residues), which are in close contact. All those atom interactions contribute to the folding of different structural motifs of neighboring residues. Common structural motifs are α-helix, β-strand or coil structures. These motifs are called secondary structure elements and usually occur various times in each protein. Proteins can consist of several peptide chains possibly linked by hydrogen bonds or sulfur bridges between cysteine residues.

2.1.2 Analysis of the MHC-peptide-TCR complex

To evaluate the characteristics of the MHC-peptide the crystal structures with or without bound ligand peptide can be examined. Furthermore, there are crystal structures available with TCR attached to the MHC-peptide complex [19]. Residue specific interactions between MHC binding pocket and antigen ligand can be compared if several crystal structures with the same type of MHC are cocrystallized with different ligand peptides. This is the case for the A*0201 subtype of HLA. To deal with different types of atom-atom interactions, atom pair distances are measured within a specific cutoff threshold and it is discriminated between partially charged polar atoms and uncharged atoms. These atom-atom contacts are classified depending on whether they involve side chain or backbone of an amino acid. Side chain atom contacts are residue type specific whereas backbone atom contacts are more unspecific to the type of amino acid.
The MHC class I of type HLA-A2 has a binding groove in the α-chain of the protein, designed to bind a polypeptide chain of eight to ten residues length. For HLA-A2 it is believed that the common binding mode allows nine residues to be bound in the binding groove, while central residues of longer chains might loop out of the pocket such that terminal residues are covered completely in the binding groove of HLA-A2. The binding groove is flanked by two α-helices, one on each side, while the bottom of the groove is limited by a β-sheet structure composed of antiparallel beta strands (as shown in fig. 2.3). It is known that the binding groove of MHC type HLA-A0201 contains six anchors positions labeled A to F [18] (see fig. 1.2 on p.5). All anchors are located at the junction of the β-sheet and an α helix (B through E) or between the two α-helices (A and F). Anchor postions B and F are responsible of binding HLA-A0201 key residues 2 and 9 [20]. Key residues are those sequence positions where the amino acid type is highly conserved, such that it can be assumed that these residue types are important for the

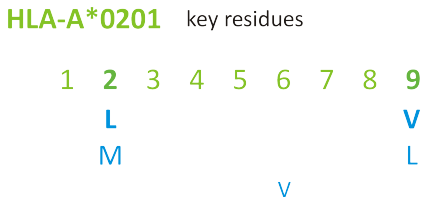


Figure 2.2: Highly conserved key positions for HLA-A*0201 are position 2 and 9.

binding of the peptides to the MHC. Accordingly at position 2 and 9 HLA-A*0201 antigens contain preferably hydrophobic residues like Leucin and Valine.

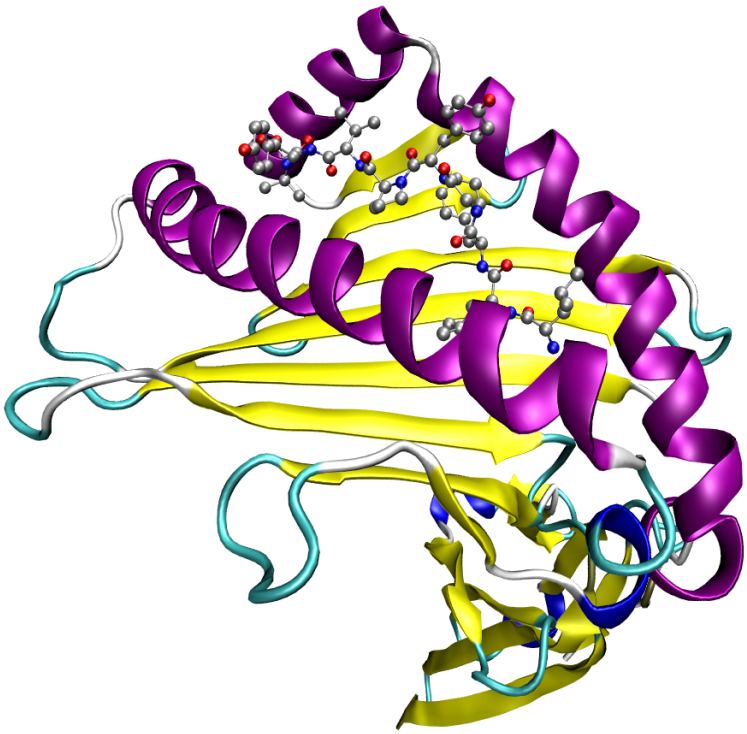


Figure 2.3: Section of the α-chain of MHC HLA-A*0201 showing the binding pocket with bound peptide (ball-and-stick model) *(from 1AO7)*

2.2 Material

2.2.1 Acquiring Antigen peptides for HLA-A0201

Knowledge based classification methods require a number of positive and negative examples to train the system. For binding predictions this means it is required to have data of ligands known to bind and data of poor binders. For the present work two available databases of MHC ligands (epitopes) for different MHC/HLA receptors were used: SYFPEITHI [2] and MHCPEP [3]. For HLA-A0201 those antigen peptides were selected consisting of exactly nine residues to be comparable. All unique sequences of those two databases were combined to one file of HLA-A0201 binders. Unfortunately I was initially not able to find any data of ligands with a weak binding affinity. In the meantime, those databases are available [12]. Two different strategies were used to derive sets of non-binding peptides. From the sequence pool of known proteins of all different kinds, random sequence fragments of nona-peptides were generated. Out of those peptide sequences known ligands with affinity to A0201 are discarded. The remaining random sequences may contain a certain percentage of binding antigens, but their number is negligible compared to the total number of sequences. In principle, this method can generate a very large number of pseudo non-binders. The second strategy was to use HIV-protein sequences, which have been used for screening experiments to understand which sequence fragments bind to which

type of HLA molecules. If all sequence fragments found to bind HLA-A0201 are removed from the sequences of the proteins, the remaining sequences can be used to deduce true non-binding nona-peptides as described above. Using this approach, the total number of possible non-binding sequences is low compared to the first method of artificial generation of non-binders and the sequence space considered is more incomplete but all resulting sequences are true non-binders.
In September 2003 268 antigen peptides for A0201 were listed in SYFPEITHI where 204 of those sequences possessed the canonical length of 9 residues. All SYFPEITHI peptides were aligned in the way they bind to the MHC binding site. All remaining 64 antigen peptides of a length longer than 9 residues were chopped to the appropriate length of nine residues such that their alignment matched with those regular nona-peptides for the MHC binding site. The MHCPEP database contained 506 antigen peptides to bind A0201. In difference to SYFPEITHI, those peptides were not aligned such that only nona-peptides were considered. Merging the two sets of antigen peptides and removing identical peptides from the set the total binding set $\mathbb{S}^+$ yielded 538 peptides (listed in Appendix table A.1).
Since no database lists explicit non-binding peptides for MHC subtypes, randomly chosen nona-peptides were used assuming that they are unlikely to bind A0201. For the non-binding set $\mathbb{S}^-$ 10,000 different nona-peptides were randomly chosen from concatenated sequences of 202 proteins selected from the protein database (see Appendix table A.2). All known binding sequences for A0201 were removed. The occurrence probability for all 20 amino acid types in the sequence data of all designated 10,000 non-binding peptides is similar to the distribution of other sequence databases containing protein sequences of vertebrates [21], but differs in some amino acid types (Ala, Arg, Asp, Glu, Leu, Lys, Val) (see table 2.1).

amino acid type	**in modern vertebrates** [1]	**non-binding peptides** [2]	**binding peptides** [3]	**amino acid type**	**in modern vertebrates**	**non-binding peptides**	**binding peptides**
Ala	0.078	0.074	0.110	**Leu**	0.089	0.086	0.180
Arg	0.063	0.047	0.027	**Lys**	0.078	0.060	0.038
Asn	0.034	0.048	0.027	**Met**	0.024	0.020	0.024
Asp	0.054	0.056	0.027	**Phe**	0.036	0.042	0.046
Cys	0.008	0.018	0.013	**Pro**	0.044	0.048	0.046
Gln	0.032	0.039	0.027	**Ser**	0.047	0.066	0.054
Glu	0.086	0.064	0.039	**Thr**	0.049	0.058	0.048
Gly	0.073	0.076	0.063	**Trp**	0.010	0.017	0.013
His	0.019	0.024	0.019	**Tyr**	0.030	0.037	0.025
Ile	0.067	0.055	0.067	**Val**	0.082	0.066	0.110

Table 2.1: Amino acid probability distributions for different sets

One limitation of these data presented by the databases SYFPEITHI and MHCPEP is a very coarse separation of binders to nonbinders. There are no binding constants listed for the antigen peptides, but yet they are classified as binding peptides. It is known that entries added to the database are taken from literature, mostly results from experimental research of binding studies. This offers another possibility of acquiring data for binding and non-binding peptides for A0201. Some groups have published results of binding studies providing binding constants [16][22].

[1] Probability of occurrence of amino acid types in modern vertebrates according to Ref. (25).
[2] Probability of occurrence of amino acid types in the 10,000 non-binding nona-peptides of set $\mathbb{S}^-$
[3] Probability of occurrence of amino acid types in the 538 binding nona-peptides of set $\mathbb{S}^+$

2.2.2 Crystal structures used for structure comparison

To analyze and compare structures of the HLA-A*0201 complex, 14 crystal structures from the pdb database [23] were used. Five of the 14 structures were cocrystallized with bound TCR molecule and antigen nona-peptides [24, 25, 26, 19]. The remaining nine structures contain solely antigen peptides bound to the MHC binding pocket (pMHC) [27, 28, 29, 30, 31]. In three of these 9 structures antigen deca-peptides were bound, while for all other cases nona-peptides were bound. The following table 2.2 provides an overview of the employed crystal structures.
The binding pockets of the A0201 complexes with different ligand peptides were analyzed with respect to contact distances to nearby residues. The presence of hydrogen bridges, salt bridges or hydrophobic interactions were examined. To compare conformational changes of different ligands in the A0201 binding pocket, A0201 structures were superimposed such that the binding pockets overlap. Results are shown in section 3.1.

pdb ID	TCR type	source of antigen peptide	antigen peptide length	antigen peptide sequence
1AO7	A6 human	from TAX	9	LLFGYPVYV
1BD2	B7 human	from TAX	9	LLFGYPVYV
1QSF	A6 human	from TAX Y8A	9	LLFGYPVAV
1QSE	A6 human	from TAX Y8A	9	LLFGYPRYV
1LP9	AHII 12.2 mouse	P1049	9	ALWGFFPVL
1DUZ	none	from TAX	9	LLFGYPVYV
1AKJ	none	HIV-1 peptide	9	ILKEPVHGV
1QEW	none	P01884	9	FLWGPRALV
1HHG	none	HIV-1 peptide	9	TLTSCNTSV
1HHI	none	influenza pro. pep.	9	GILGFVFTL
1HHJ	none	HIV-1 peptide	9	ILKEPVHGV
2CLR	none	pep. Calreticulin	10	MLLSVPLLLG
1I4F	none	melanoma antig.4	10	GVYDGREHYV
1HHH	none	hepatitis B pep.	10	FLPSDFFPSV

Table 2.2: Crystal structures used from pdb with HLA-A*0201 and antigen ligand bound

2.2.3 CoEPrA data

The Comparative Evaluation of Prediction Algorithms 2006 (CoEPrA) [5] provides data sets for classification (and regression) of different ligand peptides. The CoEPrA competition is divided into four classification tasks. For each task one learning and one prediction data set is given. Each data set contains a binding and non-binding set, providing peptide sequence, feature vectors as molecular descriptor and class expectation value for each entry. The prediction data sets contain only sequence and feature vectors, while the class expectation values are not provided. Each residue position is described with 643 physico-chemical features, the entire peptide of n residues length therefore results in $n \cdot 643$ features. Most of the features are extracted from the AAindex database [32] (see Appendix A.1 for more information). Table 2.3 lists different classification tasks from CoEPrA 2006 together with the distribution of the data sets. All tasks except of number 4 show a symmetrical distribution between binding and non-binding data.

no. of task	type of class	length of sequence	no. of features	no. of peptides in sets learning	prediction[a]
problem 1	+	9	5787	44	44
	-	9	5787	45	44
problem 2	+	8	5144	37	38
	-	8	5144	39	38
problem 3	+	9	5787	67	67
	-	9	5787	66	66
problem 4	+	9	5787	19	19
	-	9	5787	92	92

[a]division into classes was not known in advance for the prediction set

Table 2.3: CoEPrA 2006 data sets, classification tasks 1 - 4

2.3 Methods

2.3.1 Structural analysis of HLA-A*0201 binding pocket and bound peptides

To understand the binding mode for peptide ligands in the binding pocket of A0201 all complexes with and without cocrystallized TCR are separately superimposed using the Kabsch algorithm [33][34] such that the α chains of the different structures are overlaid. Removing the HLA-A*0201 molecules leaves the bound ligands superimposed in the binding pockets of A0201. Conformational variations in the bound ligands caused by different sequences are easy to recognize. The Root Mean Square Deviation (RMSD) is a measure of conformational deviation calculated over all equivalent atoms from two given conformers $RMSD = \sqrt{\frac{1}{N}\sum_{i=0}^{N}(x_{a,i} - x_{b,i})^2}$.
Analyzing the type of atom-atom interactions between ligand and protein helps to understand sequence position specificities of certain ligand residues. Further examination can reveal how well amino acid types of a certain residue position of the ligand fit into the binding groove. A atom-atom contact is counted if any atom of a residue on the ligand is below a contact distance threshold to any other atom of a residue from the α chain. No hydrogen atoms are considered for this procedure. Depending on the type of atoms and the functional groups that form these interactions, the contact is called hydrophilic in case of a salt bridge (charge-charge interaction) or hydrogen bridge (hydrogen donor to hydrogen acceptor groups) or it is considered hydrophobic in all other cases. For hydrophilic and hydrophobic contacts a contact distance threshold of $d \leq 3.5$Å was set.

2.3.2 Choice of the scoring function

The basic idea is to choose a function describing the correlation between molecular descriptors (features) $\vec{x}$ and target values y indicating the class to which the molecule belong to (in this case for binding and non-binding molecules). It is assumed that

$$y = f(\vec{x}) \qquad \text{where } \vec{x} \in \mathbb{R}^n \quad \text{and } y \in \mathbb{R} \tag{2.1}$$

for a number of couples of $\vec{x}$ and y. The goal is to approximate this relation in order to predict target values y for any given feature vector $\vec{x}$.

2.3.2a Data representation

For the classification approach it can be assumed to have two classes of molecules, those which are binding (positive class/+) and those which are non-binding (negative class/-). Four sets of molecules are used, one learning and one test prediction set for each class. After the training of the method is complete the learning set will be reused for a pseudo prediction, the recall. The test set will be used for a test prediction to control the progress after learning and the capability to generalize learned patterns.
Let's assume we have two learning sets of molecules, the set of binding molecules $\mathbb{S}^+ = \{\vec{x}_n^+,\, n = 1, \dots N^+\}$ and the set of non-binding molecules $\mathbb{S}^- = \{\vec{x}_n^-,\, n = 1, \dots N^-\}$. All molecules $\vec{x}_n$ in those sets are represented by feature vectors composed of K features

$$\vec{x}_n = (x_{1,n}, x_{2,n}, \dots, x_{K,n})\,, \text{ where } x_{k,n} \in \mathbb{R} \tag{2.2}$$

In the special case where sequence information of peptides is used as features, one binary subvector is used for each residue position coding the amino acid on that position. K is the sequence length, yielding

$$\vec{x}_n^t = \left(\vec{x}_{1,n}^t, \vec{x}_{2,n}^t, \dots \vec{x}_{K,n}^t\right) \tag{2.3}$$

where each subvector in eqn. 2.3 posses 20 components

$$\vec{x}_{k,n}^t = \left(x_1^{k,n}, x_2^{k,n}, \dots x_{20}^{k,n}\right) \tag{2.4}$$

which refer to the 20 different native amino acid types. The amino acid type at a particular sequence position is coded by setting the corresponding component of the subvector to unity, while all other 19 components of this subvector contain zeros. The advantage of this representation comes to mind when interpreting each subvector as a probability distribution to find specific amino acid types at the corresponding sequence position.

2.3.2b The linear scoring function

The scoring function $f(\vec{x})$ defines whether a molecule represented by its feature vector $\vec{x}$ is classified as binder or non-binder. The bare scoring function is linear in feature space (respectively sequence space) $\mathbb{S}$. This linear form of the scoring function can be written as

$$f\left(\vec{x}\right) = \vec{w}^t \bullet \vec{x} + b \tag{2.5}$$

where $\vec{x}$ is a K component feature vector describing the particular molecule, $\vec{w}^t$ is a row vector of the same dimension as $\vec{x}$ and b is a scalar. There are $K+1$ free parameters in this scoring function given by $\vec{w}^t$ and b, which have to be determined for the set of molecules used for learning $\mathbb{S}_{learn}$ such that $f(\vec{x})$ adopts a value close to the assigned target value of $+1$ for binding molecules and close to -1 for the non-binding molecules. Hence, setting $f(\vec{x}) = 0$ defines a hyperplane in the K-dimensional feature space $\mathbb{S}$ with plane normal vector $\vec{w}$, which separates binding molecules $\vec{x}^+$ with $f\left(\vec{x}^+\right) > 0$ from non-binding sequences $\vec{x}^-$ with $f\left(\vec{x}^-\right) < 0$. These criteria can be used to predict the binding ability of molecules.

2.3.2c Least square optimization

One of the most elementary strategies to determine the $K+1$ free parameters of the scoring function $f(\vec{x})$ of eqn. 2.5 is to minimize the scoring function with respect to the sum of the least square deviations called least square method (LSM).

$$L\left(\vec{w},b\right)=\frac{1}{2\,N}\sum_{n=1}^{N}\left(f\left(\vec{x}_n\right)-y_n\right)^2 \tag{2.6}$$

The sum in eqn. 2.6 runs over all molecules of the learning set $\mathbb{S}_{learn}=\mathbb{S}^+\cup\mathbb{S}^-$, where the target values $y_n=+1$ for binding molecules and $y_n=-1$ for non-binding molecules are used. Taking the derivatives of $L\left(\vec{w},b\right)$ with respect to $\vec{w}$ and b result in the following set of K linear equations

$$\left\langle\left(\vec{x}-\langle\vec{x}\rangle\right)\left(\vec{x}^t-\langle\vec{x}^t\rangle\right)\right\rangle\bullet\vec{w}=\left\langle\left(y-\langle y\rangle\right)\left(\vec{x}-\langle\vec{x}\rangle\right)\right\rangle \tag{2.7}$$

and

$$b=\langle y\rangle-\langle\vec{x}^t\rangle\bullet\vec{w}\,. \tag{2.8}$$

The angular brackets in these equations denote averages over all molecules of the learning set $\mathbb{S}_{learn}$ as for instance

$$\langle\vec{x}\rangle=\frac{1}{N}\sum_{n=1}^{N}\vec{x}_n\,. \tag{2.9}$$

2.3.2d Weighting and regularization

To obtain higher flexibility it is possible to split the terms for binding and non-binding molecules and weight them differently. Using equation 2.9 this would lead to

$$\langle\vec{x}\rangle=\frac{w^+}{N^+}\sum_{n=1}^{N+}\vec{x}_n^+ +\frac{w-}{N^-}\sum_{n=1}^{N^-}\vec{x}_n^-\,, \tag{2.10}$$

where $w^+ + w^- = 1$ and $N^+ + N^- = N$. This description allows a weighting of molecules from the learning set $\mathbb{S}_{learn}$, which is independent from the actual number of binding and non-binding molecules in this set. For most cases best results arise for weighting factors w^+ close to 0.50.
It is known that large number of free parameters along with a small number of molecules in the training set will lead to overfitting, which is causing the learning by heart phenomenon. This leads to bad generalization capabilities and a collapse in prediction quality, while the recognition rate of learned data becomes even higher. One solution of this problem is called ridge regression [35] and is often used if two highly self correlated predictor variables are used for least square optimization because the derived coefficients may be imprecise. The ridge regression method adds up small fixed values to the diagonal elements of the coefficient matrix. This is introducing a certain bias, but suppresses numerical instability. Here it can be understood as introduction of an additional regularization term:

$$\tilde{L}\left(\vec{w},b\right)=L\left(\vec{w},b\right)+\lambda\,\vec{w}^t\bullet\vec{w}\,, \tag{2.11}$$

where λ is an empirical parameter to be chosen. Due to the normalization of the optimization function $L\left(\vec{w},b\right)$ in eqn. 2.6 by the number of molecules in the learning set N, the value of λ is

independent of the learning set size. The regularization term avoids the occurrence of singular behavior and contributes to a minimization of the length of the normal vector $\vec{w}$ of the separating hyperplane that is defined by $f(\vec{x}) = \vec{w}^t \cdot \vec{x} + b = 0$. As a consequence, the sensitivity of this separating hyperplane may increase for moderate values of λ. This is the case in particular if the set of linear equations 2.7 is ill-conditioned due to the smallness of the learning set $\mathbb{S}_{learn}$. Applied to the set of linear equations the expression in eqn. 2.7 becomes

$$\langle (\vec{x} - \langle \vec{x} \rangle) \left(\vec{x}^t - \langle \vec{x}^t \rangle \right) \rangle \bullet \vec{w} + \lambda \vec{w} = \langle (y - \langle y \rangle) (\vec{x} - \langle \vec{x} \rangle) \rangle \,. \tag{2.12}$$

In most examined cases a value for λ of 10^{-10} is large enough to suppress singular behavior, but may still be too small to cause a negative bias for prediction quality.

2.3.3 Quadratic scoring function

With minor modifications the scoring function can be applied also with quadratic feature terms derived as products of linear features. Since the scoring function calculates the covariance between features for linear feature terms, second order feature correlations are already used in case of a linear scoring function. Thus the use of quadratic feature terms leads to the use of fourth order feature correlation terms. One can rewrite the scoring function from eqn. 2.5 such that the quadratic terms fit into the linear equation scheme.

$$y = f(\vec{x}) = F\left(\vec{X}\right) = \vec{V} \cdot \vec{X} + b \tag{2.13}$$

where

$$\begin{array}{rcl} & & \underline{\quad linear \quad} \quad \underline{\qquad quadratic \qquad} \\ \vec{X} & = & \{ \quad x_0, x_1, \ldots, x_K, \quad \tilde{x}_0\,\tilde{x}_0, \tilde{x}_0\,\tilde{x}_1, \ldots, \tilde{x}_K\,\tilde{x}_K \quad \} \\ \vec{V} & = & \{ \quad w_0, w_1, \ldots, w_K, \quad W_{00}, 2W_{01}, \ldots, W_{KK} \quad \} \end{array}$$

Alternatively one can manually derive quadratic feature terms by calculating the product of two given linear feature terms and treat them as new linear feature. This approach is most flexible to derive only those single quadratic features that correspond to selected pairs of linear feature terms.

2.3.4 Cholesky versus LU decomposition

To solve the linear equation systems defined by the coefficient matrix A of the type

$$A \cdot \vec{x} = \vec{y} \tag{2.14}$$

the LU decomposition, a modified form of the gaussian elimination, is used. The matrix A shall be decomposed to an upper and a lower triangular matrix

$$A = LU \tag{2.15}$$

which contain zeros below or above the matrix diagonal respectively.
In case where the matrix A is positive definite one can use the Cholesky decomposition instead, named after André-Louis Cholesky. The definition is

$$A = L\,L^T, \tag{2.16}$$

where L^T is the transpose of the lower triangular matrix L. The Cholesky decomposition is faster, because the number of required operations is by a factor of 2 lower compared to LU decomposition. The Cholesky method is numerically more stable and doesn't require pivoting at all in contrast to LU decomposition. In this case the Cholesky decomposition was applicable and used, because A is positive definite and symmetric.

2.3.5 Protocol describing the prediction strategy

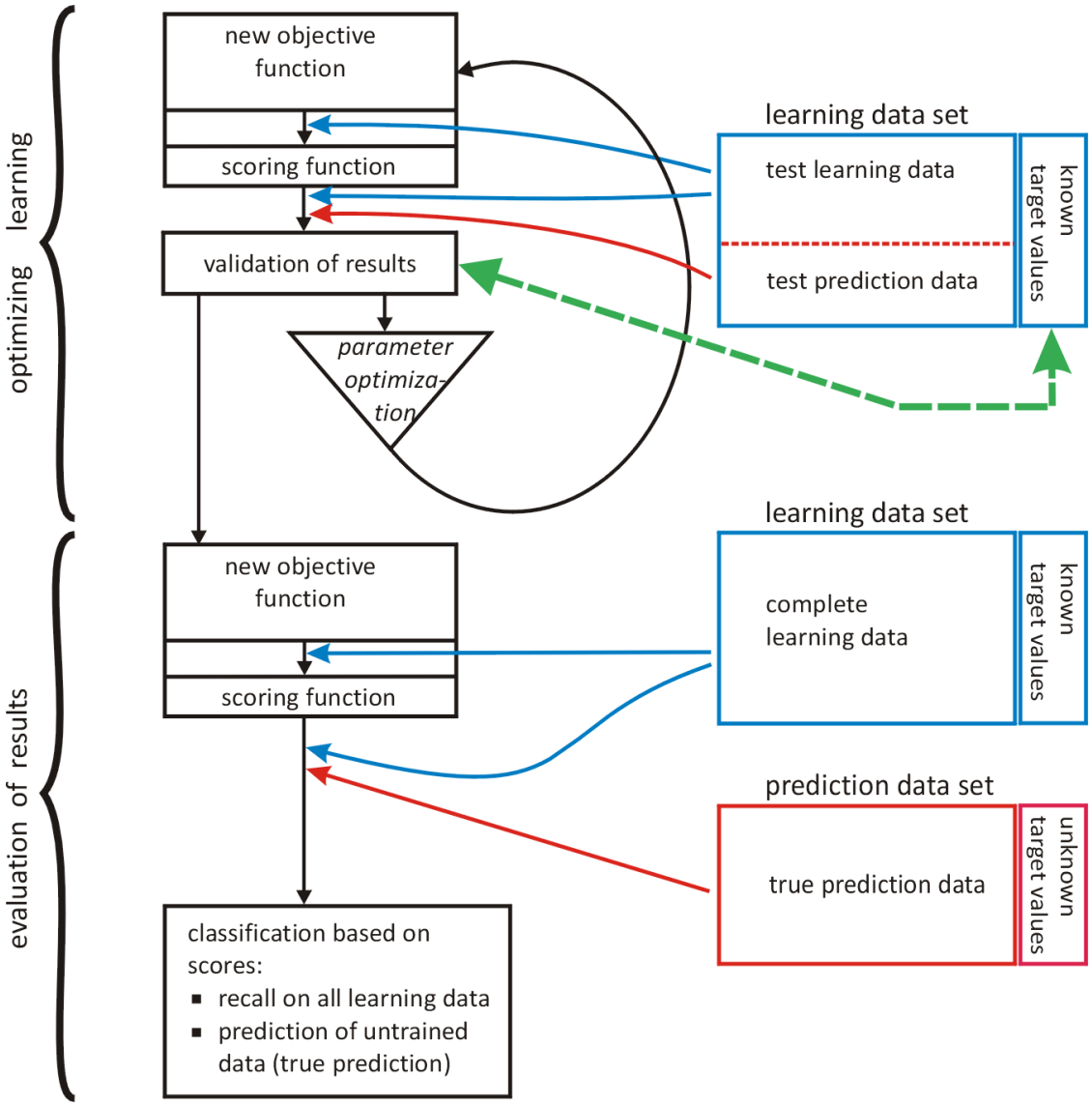


Figure 2.4: Prediction procedure using fixed features.

For true predictions the target values of the prediction data set are unknown. Therefore optimizing classification depends on the learning set, where target values are available. To tune the parameters of the objective and scoring function knowledge on how the method is able to generalize learned feature characteristics to untrained data is needed. Therefore, it is required to divide the set of learning data into two subsets, a true training data set and a test prediction data set. Only the training data set is used for learning, while the test prediction data set is used for pseudo prediction. Pseudo prediction means that the target values of the molecules are actually known, but the feature vectors have not been used to correlate with the target values in the learning process. This quality control helps to detect cases of learning by heart and therefore the feedback can be used to optimize free parameters $(\lambda_w, w^+, \ldots)$ of the objective and the scoring function. Another optimization possibility is given by the number of features used to describe

molecules from the different data sets. Their number should be correlated to the number of molecules in the learning data set. If the number of features is large compared to the number of given molecules in the learning data set, overfitting will occur leading to learning by heart. The opposite case is underfitting caused by using not enough features in the molecule descriptor. In this case neither the learning nor the test prediction data are classified reliably. Figure 2.4 shows the protocol for prediction strategy using a fixed set of features.
The procedure for dealing with variable feature sets is explained in subsection 2.3.11 describing the genetic algorithm.

2.3.6 Support Vector Machine (SVM)

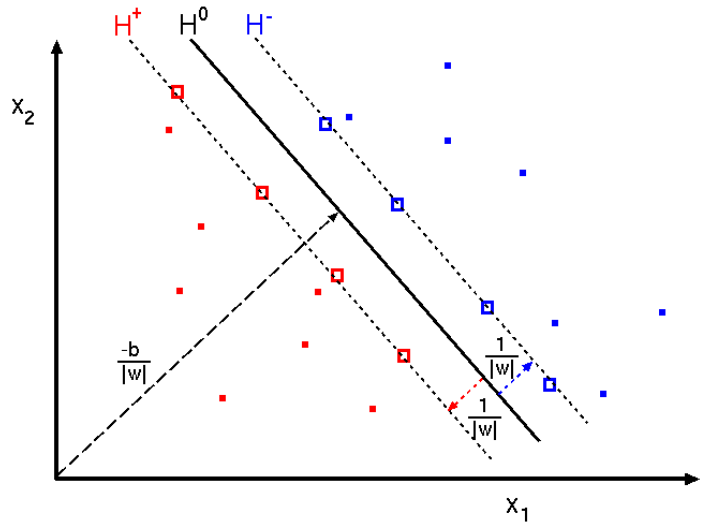


Figure 2.5: Two groups of data points (red or blue) in multidimensional feature space are separated by hyperplane H^0. Support vectors are taken from those data points lying between the parallels H^+ and H^- and the hyperplane.

For comparison another well established optimization method, the support vector machine, was used to confront it with the least square optimization used in the present work.
The support vector machine is a classification approach using a hyperplane to separate data points $\vec{x}_i$ in a n-dimensional feature space. Every object (molecules in our case) is defined by a vector and assigned to a class y_i, which is basically +1 or -1. The method employs only those objects, which are located close to the separating hyperplane. Their vectors will be accounted as support vectors, giving them a direct influence to the parameters defining the location of the hyperplane.
Given is a leaning data set Φ:

$$\Phi = \{(\vec{x}_i, y_i) \,|x_i \in \mathbb{R}^n, y_i \in \{\pm 1\}, i = 1, \ldots, L\} \tag{2.17}$$

The function $y = f(\vec{x})$ as linear approach is approximated by

$$f(\vec{x}) = \vec{w}^t \cdot \vec{x} + b \tag{2.18}$$

such that

$$\overline{f}(\vec{x}) = \text{sign}\left(\vec{w}^t \cdot \vec{x} + b\right), \tag{2.19}$$

yielding the values $\overline{f}(\vec{x}) = \pm 1$. Those vectors $\vec{x} \in \mathbb{R}^n$ that fullfill eqn. 2.18 define the hyperplane H^0 of the dimension $n-1$ in $\mathbb{R}^n$. The normal vector of the hyperplane is $\vec{w}$ and its' distance from the origin is $-b/|\vec{w}|$.
It is possible to extend the linear form of the support vector machine such that it uses non-linear kernels replacing the expression $\vec{w}^t \cdot \vec{x}$ by a polynomial, gaussian or radial basis function.
The program SVM-light 5.0 [36, 37, 38, 39] from Thorsten Joachims was used to evaluate the method of support vector machines.

2.3.7 Quality measure: Matthews Correlation Coefficient (MCC)

The difficulty to express prediction quality with a single value is related to the problem to describe four different criteria by a single valued quantity. For a two class classification problem the numbers of true positives (TP), true negatives (TN), false positives (FP) and false negative (FN) characterize the performance of a prediction. If only percentages of correct recognized/predicted items are provided, the number of misclassified items actually belonging to the opposite class is neglected. Thus one needs in a two-class problem to provide both, the percentage of correct classified items of class 1 and 2. Usually the average of both percentages is calculated to provide some measure of prediction quality. The 2x2 contingency matrix faces predicted to observed results as shown in the following table:

		prediction	
		class +	class -
observation	class +	TP	FN
	class -	FP	TN

A general method of estimating the quality of prediction results was introduced by B.W. Matthews in 1975 [40] providing an index to correlate prediction and observation. The correlation coefficient varies from -1 to +1. The value of +1 means a perfect match while -1 is a complete opposite match. A value of zero corresponds to completely uncorrelated, random results. In the common case of secondary structure prediction P_n represents the prediction for a given residue n of a polypeptide chain. It could be 1 for the secondary structure, if residue is related to helix or 0 otherwise. Alternatively P_n could also represent a probability distribution. The symbol S_n represents therefore the observed state of the given residue n. The correlation is given by the following formula:

$$C = \frac{\sum_n \left(S_n - \overline{S}\right)\left(P_n - \overline{P}\right)}{\sqrt{\sum_n \left(S_n - \overline{S}\right)^2 \sum_n \left(P_n - \overline{P}\right)^2}} \tag{2.20}$$

$\overline{S}$ and $\overline{P}$ are mean values over all N residues. The denominator holds the product of the standard deviations of the observed and the predicted assignments. For special case that values of P and S can only become zero or unity and behave like a step function the formula becomes:

$$C = \frac{p/N - \overline{P}\,\overline{S}}{\sqrt{\overline{P}\,\overline{S}\left(1-\overline{S}\right)\left(1-\overline{P}\right)}} \tag{2.21}$$

Here $\overline{S} = (p+u)/N$ and $\overline{P} = (p+v)/N$. The quantities p, q, u and v are the number of true positive, true negative, false negative and false positive assignments, which are directly

correlating prediction results with observation (expected) results.

$$C = \frac{p - N^2 \overline{P}\,\overline{S}}{N\sqrt{\overline{P}\,\overline{S}\,\left(1-\overline{S}\right)\,\left(1-\overline{P}\right)}} \tag{2.22}$$

$$= \frac{N\,p - (p+u)\,(p+v)}{\sqrt{(p+u)\,(p+v)\,(q+u)\,(q+v)}} \tag{2.23}$$

which finally leads us to the commonly used form of the MCC:

$$C = MCC = \frac{(p\,q) - (u\,v)}{\sqrt{(p+u)\,(p+v)\,(q+u)\,(q+v)}} \tag{2.24}$$

The MCC coefficient uses all four criteria (p,q,u,v) and may provide a more balanced quality measure for predictions as simple percentages can. Nevertheless, there are situations, where the MCC provides unfair judgment. This is for instance the case, if the number of false positives is very low or zero and at the same time the number of true positives is low.

2.3.8 Feature Normalization

Feature vectors derived from peptide sequences, which are described as binary position vectors are normalized by division with their sequence length. Contrary to that feature vectors assembled out of physico-chemical properties require a normalization, because the single feature values can differ in magnitude and sign as usually floating point values are used. Two different kinds of feature vectors have to be distinguished.Those vectors, which are given for each molecule in the data set running over all feature terms of the molecule (the row vectors in our representation) and those feature vectors, which are given for each property of a molecular descriptor and running over all molecules of the learning (or prediction) set (the column vectors).

To regularize features f_k^n for all given N_{Learn} molecules each of the K individual features of the molecular descriptor should be considered separately, independent whether they are derived as product of two features or not. For regularization the mean values

$$\langle f_k \rangle_N = \frac{1}{N_{learn}} \sum_{n=1}^{N_{learn}} f_k^{(n)} \qquad k = 1, 2, \ldots K \tag{2.25}$$

are subtracted to obtain the regularized features $\tilde{f_k}^n$

$$\tilde{f_k}^{(n)} = f_k^{(n)} - \langle f_k \rangle_N \tag{2.26}$$

During this procedure all those features with vanishing mean values for all molecules N are eliminated since they are not useful for the classification. For simplicity the tilde for normalized features is omitted in the following. Averages for molecules of the binding or non-binding class are discriminated:

$$\langle f_k \rangle_{N+} = \frac{1}{N+} \sum_{n+=1}^{N^+_{learn}} f_k^{n+} \qquad n+ \in \mathbb{S}^+ \tag{2.27}$$

$$\langle f_k \rangle_{N-} = \frac{1}{N-} \sum_{n-=1}^{N^-_{learn}} f_k^{n-} \qquad n- \in \mathbb{S}^-$$

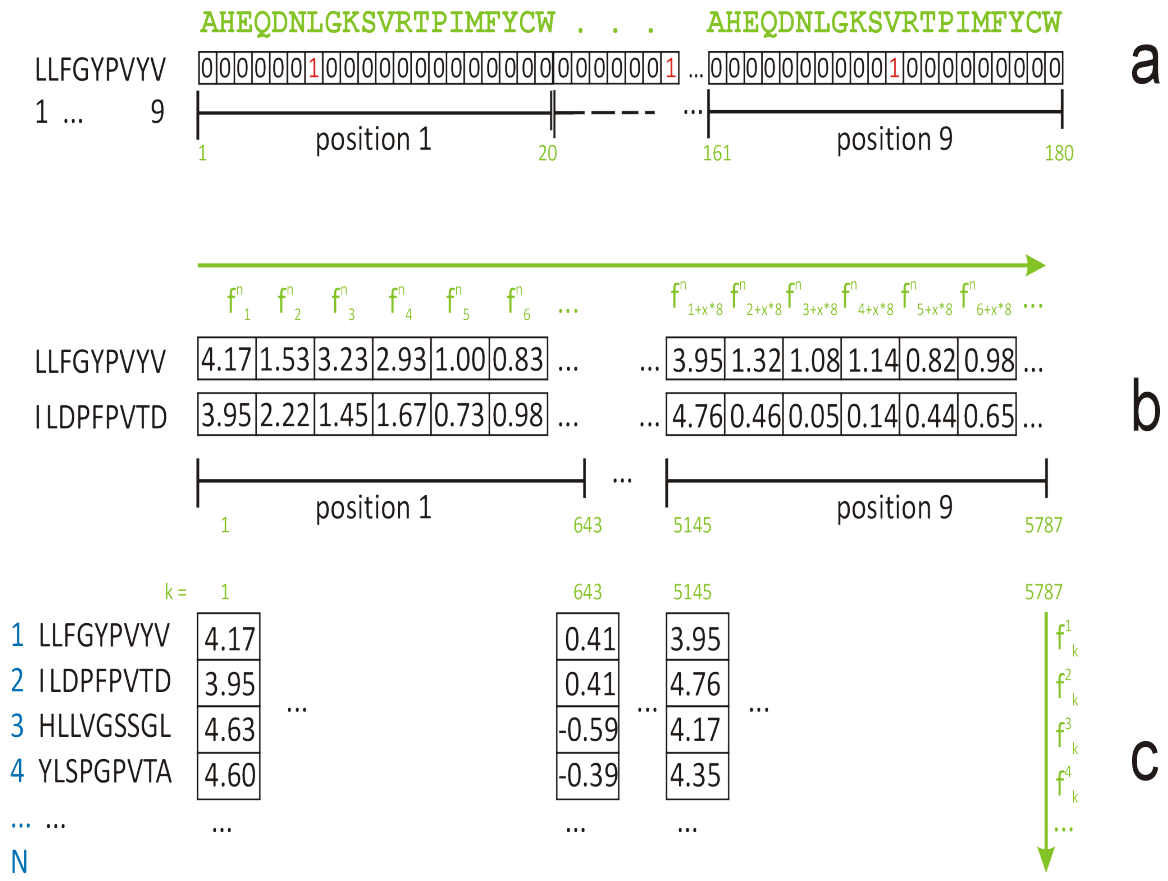


Figure 2.6: Different types of feature vectors: **a)** binary sequence vectors coding amino acid types in position dependent vectors **b)** molecular feature vectors running over all features of a molecular descriptor of single molecules **c)** feature type feature vectors running over all molecules of the data set for just the same feature type

where $n+$ runs over all molecules of the binding set and $n-$ over all molecules of the non-binding set. This leads to a regularization of f_k

$$\hat{f}_k^{\,n} = f_k^n - \langle f_k \rangle_{N+} - \langle f_k \rangle_{N-} \qquad n \in \mathbb{S} = \mathbb{S}^+ \cup \mathbb{S}^- \tag{2.28}$$

This type of regularization is used when performance of single features is considered as explained in section 2.3.9b. Other regularization methods, as for instance dividing the features by the variance seemed not to have any impact on results.
The feature vectors running over all features of a molecular descriptor for each molecule $n = 1, \ldots, N$ should be normalized to become vectors of unity length

$$\langle f_0^n \rangle_K = \frac{f_k^n}{|f_k^n|}. \tag{2.29}$$

The sequence feature vectors can be normalized by division with sequence length K.

2.3.9 Feature Reduction

With quadratic features correlations of promissing linear features can be introduced to advance prediction capability. Beside of increasing the number of qualitative good feature candidates, the total number of features and thus the number of free parameters is increased by the power of two, while the number of classification data for training remain the same. Another problem caused by increasing the number of features is the performance of the classifying algorithm and the demand of resources to handle these features. With respect to the learning by heart problem, one can use the λ_w parameter introduced in equation 2.11 to suppress the number of free parameters unspecifically. This can be useful, if all features used are of similar quality to stabilize the linear equation system numerically. A more specific way to remove useless or bad behaving features is crucial.
Feature reduction shall reduce the number of features to a number of core features, which contribute to a high prediction performance. The optimal solution is to select just those few features, which add up to a high prediction performance, but it is difficult to find out how many features are needed. Another approach is to start with a larger number of features as required and to analyze the eigenvalues and eigenvectors of the coefficient matrix of the linear equation system. This approach is realized by the Principle Component Analysis (or Singular Value Decomposition). Depending on the size of the eigenvalues it can be decided to remove or weaken the influence of the appropriate eigenvector components before solving the linear equation system.

2.3.9a Principle Component Analysis (PCA)

PCA transforms multidimensional, correlated data provided by the coefficient matrix to a new coordinate system by applying orthogonal linear transformations. It uses eigenvectors obtained from the covariance matrix to identify the independent axis of the data. The deviations from the mean values are calculated to derive the covariance matrix and finally to compute the eigenvalue and eigenvectors out of it. The eigenvalues and the corresponding eigenvectors are sorted in ascending order. The largest eigenvalues correspond to eigenvectors that define the directions of the largest variance of the data. The goal is to suppress the contribution of those components giving small eigenvalues, while accounting for those with large eigenvalues, which are supposed to be significant for the classification approach.
The linear equation system looks like $A \cdot \vec{x} = \vec{y}$. To solve for

$$\vec{x} = A^{-1} \cdot \vec{y} \tag{2.30}$$

by diagonalizing

$$A_D = D^T \cdot A \cdot D \tag{2.31}$$

the resulting eigenvalues are ordered by size

$$(A_D)_{i,j} = \delta_{i,j} a_i \qquad \text{with } i,j = 1,\dots N \quad a_1 \leq a_2 \dots a_{N-1} \leq a_N \tag{2.32}$$

with the eigenvectors $\vec{d_i}$

$$D = \left(\vec{d_1}, \dots \vec{d_N}\right) \tag{2.33}$$

leading to

$$(\vec{x}_m) = \sum_{k=1}^{K<N} \left(\vec{d_k}\right)_m \left(\frac{1}{a_k}\right) \left(\vec{d_k} \cdot \vec{y}\right). \tag{2.34}$$

The sum over n starts from the largest eigenvalue a_1.
To compute the diagonal matrix D the Eispack routines [41] tred2 and tql2 were used. Tred2 is tridiagonalizing the real symmetric matrix A to compute an orthonormal and a triangular matrix using householder reductions. The tql2 subroutine is diagonalizing the given triangular matrix of tred2.
To suppress components with a small eigenvalue the term of $\left(\frac{1}{a_k}\right)$ from equation 2.34 will be used to weaken the influence the appropriate eigenvectors. One has several choices to weaken the influence of unwanted components:

1. step function behavior: all components from a given threshold t are set to 0.

2. linear decay: the components between t_1 and t_2 are faded using a linear decay down to 0.

3. nonlinear functions: fade the components between t_1 and t_2 using a function that decays non-linearly to 0.

2.3.9b Single Feature Performance

For classification it is possible to use just one single feature f_k, although this is not very effective. Depending on the ability to recognize molecules from the training set, single features can be ranked by their recognition quality. It is crucial to normalize features before analyzing them. After applying regularization to a given feature f_k according to eqn. 2.28 a molecule n from the learning set is recognized correct if

$$f_k^{n+} > 0 \qquad n+ \in \mathbb{S}^+ \tag{2.35a}$$
$$f_k^{n-} < 0 \qquad n- \in \mathbb{S}^- \tag{2.35b}$$

is true. The scoring function is establishing pair correlations between different linear features. Thus one should consider feature products (quadratic features) also for single feature performance analysis. The numbers of correctly recognized molecules from either set $\mathbb{S}^+$ and $\mathbb{S}^-$ are counted with $n^+_{correct}$ or $n^-_{correct}$ respectively. The total number of correct classified molecules from the learning set is given by $n_{correct} = n^+_{correct} + n^-_{correct}$. In the following the lower index "correct" is abbreviated by "cor.". A reliable feature k should recognize significantly more than 50% of the molecules, because a 50% recognition rate can be statistically expected from a random guess. A feature below an recognition rate of 50% can be inverted such that the rate becomes $100\% - x\%$. A quality cut-off value is defining which features are discarded from the feature set. Features of which the recognition rates are above the quality cut-off are assigned to the new feature set.
One could argue that by simply combining best ranked single features to a new feature set recognition and prediction performance would be improved, but it is not that simple. If two states of recognition are allowed for each feature per molecule, say -1 for wrong classified and 1 for correct classified, an N-dimensional decision space has to be considered, where N is the number of molecules in the learning set. In combination with each other, features can agree for the classification of some molecules, but they also can contradict each other. In the real feature space of the objective function, features can adopt different values besides from 0 and 1 such that results from feature combinations are even harder to predict. Since optimal combinations of features cannot easily be calculated, an heuristic algorithms could be used to enrich the quality of feature sets step-by-step. Therefore, a genetic algorithm is introduced to find optimized feature sets (see section 2.3.11).

2.3.9c Introducing feature groups

For the following analysis it turned out to be helpful to group the examined single features with respect to their recognition performance into three subset categories. Depending on their ability to recognize molecules of the binding or non-binding set better they are assigned to the feature groups F^+, F^- or F^0. A feature belongs to:

- F^0 group, if the number of correct recognized binders and nonbinders is almost in the same range, defined by an upper and lower threshold, allowing the number of correct binders or nonbinders to deviate from equilibrium.
- F^+ group, if the number of correct recognized binders is larger than the number of correct recognized nonbinders (and if not assigned to F^0).
- F^- group, if the number of correct recognized nonbinders is larger than the number of correct recognized binders (and if not assigned to F^0).

Mathematically it can be defined:

$$F^+ : \quad n^+_{cor.} > n^-_{cor.} + \alpha^+ \left(n^+_{cor.} + n^-_{cor.}\right) \tag{2.36a}$$

$$F^- : \quad n^-_{cor.} > n^+_{cor.} + \alpha^- \left(n^+_{cor.} + n^-_{cor.}\right) \tag{2.36b}$$

where $n^+_{cor.}$ is the number of correct binders and $n^-_{cor.}$ is the number of correct nonbinders. The parameters α^+ and α^- are upper and lower threshold values to shift the equilibrium for the remaining F^0 group, which holds the remaining features. Instead of absolute numbers like $n^+_{cor.}$ and $n^-_{cor.}$ percentages can be used. The α-thresholds should be used to adapt the size of the feature groups such that all three groups have roughly the same size. Features in all three feature groups are sorted by the total amount of correct recognized molecules $n_{cor.}$.

2.3.10 Antipode Algorithm

The number of linear features can easily reach an order of magnitude of 10^3 to 10^4 like for the CoEPrA tasks. Using quadratic features expand the number of features to 10^7 or more. For any heuristic algorithm used to create suitable feature sets this number is much too large. Feature reduction without loosing much information and predictive power is crucial. The so called antipode algorithm can reduce the number of features by eliminating features from the set, which are too similar to other features, regarding their recognition pattern. The term "antipode" characterizes objects, which are as dissimilar to each other as possible like two points being diametrically opposed to each other. The similarity between features is determined on the basis of their recognition vectors $\vec{b}_i$. This is a binary vector of each feature running over all molecules of the learning set, containing "1" for each correct recognized molecule (no matter if from the $\mathbb{S}^+$ or $\mathbb{S}^-$ set) and "-1" for each wrong classified molecule (see eqn. 2.35a and eqn. 2.35b).

$$\vec{b}_k = \left(b_k^{(1)}, b_k^{(2)}, \ldots, b_k^{(N)}\right) \quad \text{with } b_k^{(n)} \in \{-1, 1\} \tag{2.37}$$

Due to this kind of representation different features may possess the same recognition pattern and thereby the same recognition vector.

The similarity of two features i and j is given by the scalar product of the two corresponding recognition vectors

$$S_{i,j} = \frac{\vec{b}_i \cdot \vec{b}_j}{N}. \tag{2.38a}$$

$$D_{i,j} = 1 - S_{i,j} \tag{2.38b}$$

The scalar product $S_{i,j}$ varies between +1 and -1, where +1 means identity and -1 complementary of the compared recognition vectors. Analog to the similarity distance S we can define the dissimilarity measure D. The following steps of the antipode algorithm are applied for all three feature groups F^+, F^-, F^0 separately. The symbol $F^\#$ denotes a universal feature group placeholder.

1. (a) First a similarity threshold value for the given feature group $F^\#$ of $\tau_\#$ has to be defined.

   (b) Features in feature group $F^\#$ are ordered by decreasing recognition performance as described in section 2.3.9c. The first feature from $F^\#$ is picked to become the first member of the initially empty antipode feature groups $F^\#_{AP}$. Usually this is the first ranked (and best) feature of $F^\#$.

2. For the next feature to become a member of the new reduced set of $F^\#_{AP}$ a new feature j from the original list of $F^\#$ is compared with those features of the new target feature group $F^\#_{AP}$ using eqn. 2.38a. If the relation

   $$D_{i,j} > \tau_\# \qquad i \in F^\#_{AP} \quad \text{and} \quad j \in F^\# \tag{2.39}$$

   is true for all features i of the antipode set $F^\#_{AP}$, the new feature j from $F^\#$ will be added to the antipode set. The feature examination for a given feature j can be interrupted, if any feature i of the new feature group $F^\#_{AP}$ is too similar to the new feature j from the reference set $F^\#$. In that case feature j is dismissed.

3. Step 2 is repeated until all features from the reference group $F^\#$ have been examined.

It is obvious that by choosing the size of the threshold $\tau_\#$ the size of the new created feature group $F^\#_{AP}$ is influenced. The algorithm uses three different thresholds for the feature groups F^+, F^- and F^0 named τ_+, τ_- and τ_0, such that the size of all resulting antipode feature groups can be influenced separately. At the end of the antipode algorithm all three created antipode feature groups F^+_{AP}, F^-_{AP} and F^0_{AP} will become the new feature groups F^+, F^- and F^0.

2.3.11 Genetic Algorithm

For the CoEPrA competition it has been demonstrated by the participating group of Wuju Li[4] for classification problem 1 that the right choice of a few feature can be sufficient for a high ranked prediction performance. With only seven features from the CoEPrA feature set the prediction results ranked first for problem 1 with an average prediction performance of 86% correct identified molecules.
The genetic algorithm (GA) is used to generate optimized subsets of features for classification.

[4]Wuju Li, Center of Computational Biology, Beijing Institute of Basic Medical Sciences

For this kind of optimization problems, where there is no designated way to calculate exact solutions in reasonable time, heuristic algorithms based on randomization like GA are the methods of choice to find near-optimal solutions [42].
Basic principle is that a large number of initial random feature subsets is generated in the beginning. Every subset of features called an *individual* is a complete set of features used for the scoring function with a distinct solution allowing an entire classification prediction. To rank the results of all individuals, the learning set is splitted into test learning set and test prediction set. The test learning set is used to train the different feature sets provided by the individuals, while the test prediction set is used to score the performance of the trained individual. The entity of individuals achieved in each cycle of the GA is called a *generation* of individuals. In each cycle of the GA the precursor generation is modified by genetic operations. These operations change or interchange single or several features contained in the feature subset (also named *chromosome*) of a given individual by chance. While a certain percentage of top ranked individuals of the ancestor generation will be conserved for the next generation, the remaining individuals in the next generation are randomly selected or mutated individuals from the ancestor generation. With every new generation a scoring and ranking of all contained individuals will be performed. In every new cycle the overall performance of the GA should improve until it converges because no further optimization can be achieved. The scoring function used in the GA to rank the individuals is the linear scoring function introduced in section 2.3.2b. The GA is using those grouped features obtained from the antipode algorithm. Currently the total number of features per individual and the contingent of features from the different feature groups are constant parameters during the program run. Constant parameters means that their values cannot be modified during the runtime of the program and they are fixed for all individuals of all generations. One has to adopt the values of these parameters before initiating the GA.

2.3.11a Preventing learning by heart during GA

To control problems of overfitting during the optimization cycles of the GA, a switch is implemented to avoid learning by heart. If it is switched on via a given parameter, this mechanism eliminates the n best individuals of the parent generation to avoid that such individuals will dominate the following generations by its' performance caused by distinct learning by heart. In such a case the test prediction set is not diverse enough from the test learning set, since these individuals memorize patterns from test learning and test prediction set. Discarding top ranked individuals in every cycle of the GA might cause slower convergence but can in some cases help to improve the fitness of the last generation of the GA. By default this parameter, called "leave best out", is switched off.
Another mechanism which should help to prevent learning by heart is related to the distribution of molecules to test learning and test prediction set. As described in section 2.3.11e, the complete learning set is divided into quarters, where one quarter is used for test prediction, while the rest is used as test learning set. Four different scores are computed each time for the four different test prediction sets, but the molecule distribution within the quarters remains the same during the entire cycle of the current generation. This is required for comparability within one generation. After each cycle of the GA, the distribution of molecules to the quarters is changed randomly. This feature will prevent that well trained individuals get used to the distribution of the sets and therefore could memorize distribution patterns to learn by heart.

2.3.11b Genetic operations

Reproduction.
Individuals from the parent generation, which are top ranked by their scores, are preserved for the next generation to be utilized in the following cycle of the GA. The fittest individuals survive the selection process of the GA to become part of the next generation.

Remodelation. A number of individuals are generated from scratch with every new cycle of the GA. These individuals are built of randomly picked features from the feature groups. The composition of features from the different feature groups remain unchanged.

Point mutation. Individuals of the parent generation are selected by chance to be mutated in one or more feature positions of its chromosome. By random choice a feature position within the chromosome of such an individual is selected. The feature on this position of the chromosome is replaced by another feature randomly selected from the same feature group. The current number of mutations within one chromosome of the considered individual is a random choice between one and the maximum number of allowed mutations per chromosome, defined by a parameter to choose.

Recombination. Pairs of two individuals are selected by chance to perform a recombination of their randomly cut chromosomes. Both chromosomes are cut at equivalent feature position and the chromosome parts of both individuals are exchanged such that two new individuals with interchanged chromosomes are created. Both new individuals contain the same conserved composition of features from the feature groups, because the cutting position in both chromosomes is identical. This operation is also known as crossing over.

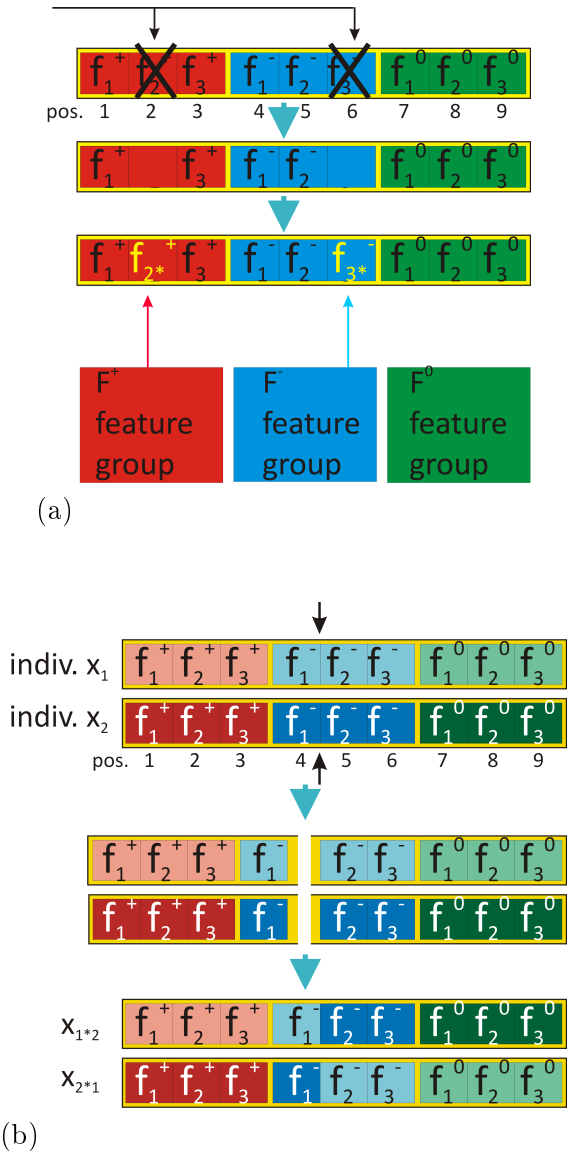


Figure 2.7: **a.** example of point mutations to the chromosome of a selected individual **b.** recombination of two chromosomes from selected individuals

The number of individuals being treated with the different genetic operations described above is given by rate-parameters to choose and the parameter defining the number of individuals per generation. First the number of individuals undergoing reproduction is set by multiplying the rate constant for recombination with the total number of individuals per generation.

$$N_{reproduce} = p_{reproduce} \cdot N_{generation}$$

Only the reproduction rate reflects the percentage of individuals to be copied to the next generation. For all other genetic operations twice as much individuals are generated in the beginning to finally cutdown the size of the desired generation size after scoring and ranking all newly generated individuals. Therefore the exact number of individuals generated by any of the genetic operations other than reproduction may vary from generation to generation and depend finally on their scoring performance.

$$\begin{aligned} \hat{N}_{pmutate} &= 2\, p_{pmutate} \cdot N_{generation} \\ \hat{N}_{recombine} &= 2\, p_{recombine} \cdot N_{generation} \\ \hat{N}_{remodulate} &= 2\, p_{remodulate} \cdot N_{generation} \\ \hat{N}_{others} &= \hat{N}_{pmutate} + \hat{N}_{recombine} + \hat{N}_{remodulate} \end{aligned}$$

$$N_{all} = N_{reproduce} + best\,of\left(\hat{N}_{others}\right)_{1\ldots(N_{generation}-Nreproduce)}$$

2.3.11c Random selection of features in the GA uses weight bias

Several operations in the GA require random feature selection from the feature groups. Instead of providing equal chances to all features during feature selection, a weighting is introduced to favor those features with a high single feature performance. This leads to a higher chance for good performing features to be selected compared to features with a weaker performance. This should improve the overall performance of the GA by reaching good results in fewer cycles.

The applied method is not directly assigning better features larger weights, but indirectly. It assumes that the number of good performing features is smaller than the number of moderate performing features. First all features of a feature group F^{+}_{AP}, F^{-}_{AP} and F^{0}_{AP} are considered separately. All features within a feature group are ranked by their recognition performance $n_{correct}$ (see 2.3.9b). Those features which have the same performance are put into the same partition. For each value of $n_{cor.}$ exists a separate partition. The random function first selects the partition by chance. All partitions have the same chance to be selected. In the second step one feature of the appropriate partition is selected by chance. Any feature in the selected partition has the same chance to be selected. Because more features are grouped in one partition assigned to a lower feature performance compared to feature entries in a high ranked partition, the chances to select a high ranked feature is increased with respect to the number of occurrence of all type of features.

2.3.11d Removing identical feature sets

There are thousands of features in each feature group and the number of feature combinations possible using just a hand full features per subset can reach a magnitude of 10^{15}. Nevertheless it is possible that individuals with identical features are created within one generation of individuals. The probability for occurrence of such doublets is increased due to the unequal weighted random

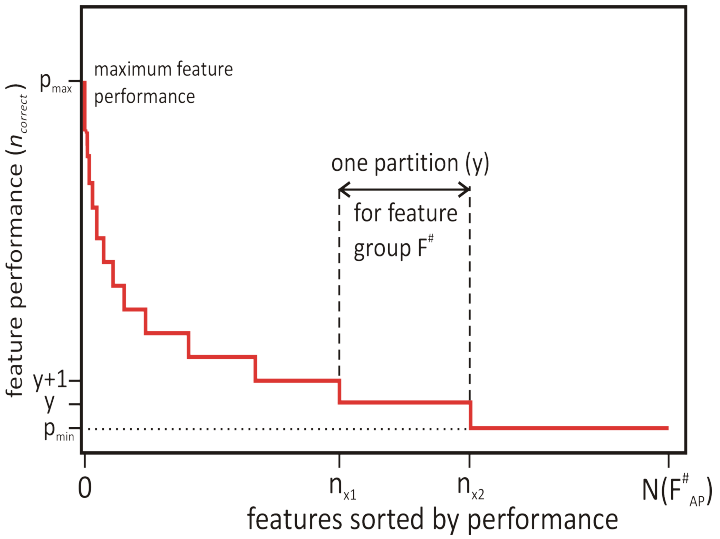


Figure 2.8: Typical example of feature performance in a feature group. Each partition, represented by a step, has the same chance to be selected.

feature selection within the GA. The optimization itself might cause the use of predominant features as a part of a repeatedly appearing pattern in a number of individuals.
The algorithm takes care of doublets before merging newly created or mutated individuals to the current generation. All newly created individuals which have clones within the current generation are eliminated such that the remaining individuals show diversity within their generation.

2.3.11e Scoring of individuals

One of the most important parts of the GA is a quality feed back realized by the implementation of the scoring function. For each individual in each cycle of the GA the scoring function has to solve the complete equation system with all free parameters. For comparison it is required that all tested individuals experience the same distribution of molecules between test learning and test prediction set.

- **Difficulties to derive meaningful distributions to learn and prediction sets**

One common problem is to find an optimal partitioning of the learning data set into test learning and test prediction set, because the dimensions of both sets will play an important role for the quality of the learning process. If the used test learning set becomes to small, the quality of learning is affected and important patterns may not be trained. On the other hand, if the test learning set is very large and the remaining part for test prediction is very small (the extreme case is jack knife or leave-one-out cross validation where only one single molecule is predicted at a time), the risk is high that learning by heart is favored. In this case the test prediction set is very similar to the larger fraction used for test learning. It turned out that a distribution of $\frac{3}{4}$ to $\frac{1}{4}$ between test learning and test prediction data set is delivering good results for complete learning data sets of a size between 50 to 150 molecules. It is important that the real prediction data set will not be used to tune the parameters. Only in this case the information obtained from a post GA analysis of the real prediction set can be used to evaluate optimization quality of the GA procedure. In a real prediction scenario the target values for the real prediction set

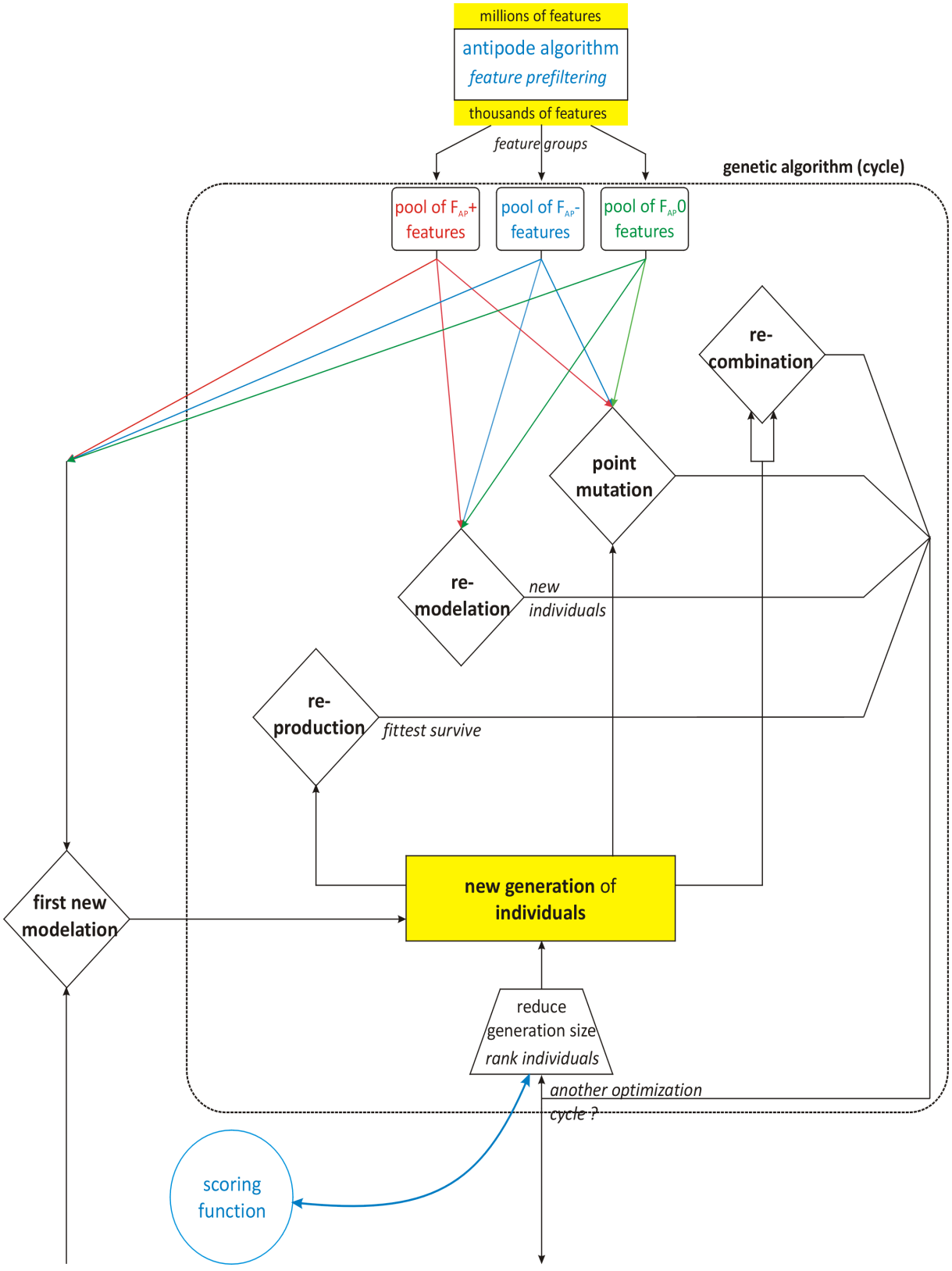


Figure 2.9: Schematic view to the genetic algorithm with its genetic operations.

are not known anyway.
The variance in learning progress can be large just by altered distribution of molecules to test learning and test prediction set. It can make a difference to shift specific molecules from test learning to test prediction set or vice versa. This is also depending on the homogeneity of the complete learning set. To minimize effects of different molecule distributions several different molecule distributions to the sets are used. For the GA the learning set is split into quarters and the scoring function is executed four times. Each quarter is becoming the test prediction set once, while the remaining three quarters are used for test learning. Averages on all scores, recall and prediction rates are calculated.

• **Obtaining the score**

The MCC is used to quantify the performance for recall (test learning) and for test prediction. The MCC values are calculated four times for each individual with different molecule distributions between the sets for the test learning set, called $MCC\,(L)$ and for the test prediction set called $MCC\,(T)$. The average values are obtained as

$$\langle MCC(aL)\rangle = \frac{1}{4}\sum_{n=1}^{4} MCC\,(L)_n \tag{2.40}$$

$$\langle MCC(aT)\rangle = \frac{1}{4}\sum_{n=1}^{4} MCC\,(T)_n \tag{2.41}$$

where the 4 reflects the four different distribution of quarter set assigned for test prediction or test learning. The aL stands for average Learning while aT means average Test prediction. The score Q used to rank individuals performance is calculated exclusively from the MCC of the test predictions

$$Q = \langle MCC(aT)\rangle + W_{min.}\, MCC(T)_{min.}. \tag{2.42}$$

The formula contains a term, which is the product of a weighting factor times the minimum MCC value of all four different test prediction. This term is summed to the average MCC value of all four test predictions. The default minimum weight used to obtain the results is $W_{min.} = 0.1$. The reason why the minimum MCC value is introduced is that the minimum reflects the reliability of the calculated average. A low minimum with respect to the average could indicate untrustworthy performance or at least show fluctuations in the performance when molecule sets are altered.
The individual score value Q is multiplied with a factor of 1000 to proceed with the bucket sort procedure [43, 44], a fast integer sortation to rank the individuals by their performance.

2.3.11f Parameters to tune the GA

There are a number of parameters of the GA, which require manual tuning to deliver satisfying results regarding different demands in classification. In this Ph.D. study the different tasks of the CoEPrA classification problems were evaluated using the GA. The table 3.29 on page 83 in the chapter "Results" gives an overview of the different parameter values applied for the different problems.

2.3.12 Post-processing

The genetic algorithm is deriving a number of individuals in the final generation, which should be enriched with good performing candidates. Nevertheless it turned out that with varying per-

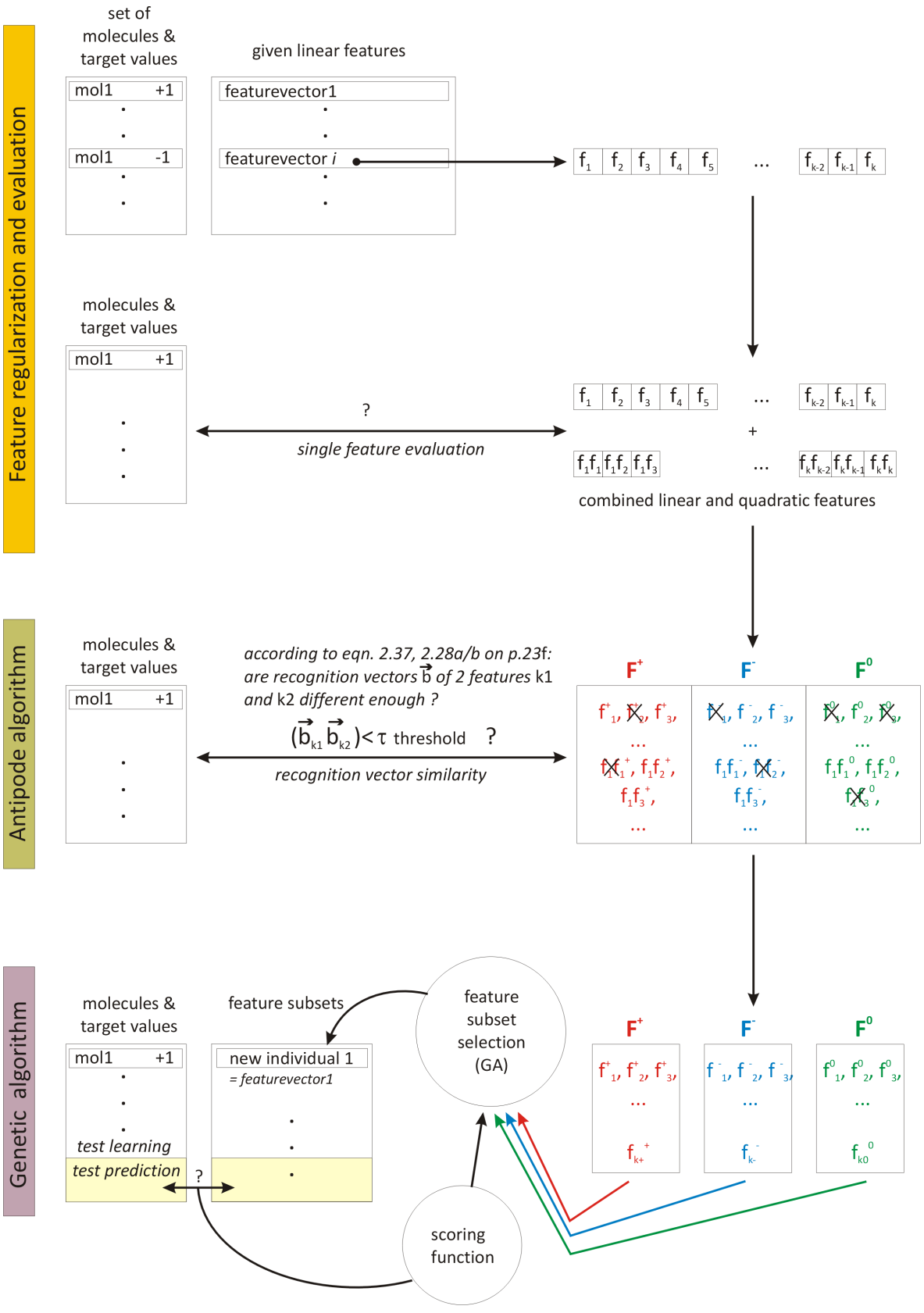


Figure 2.10: Overview of the complete procedure to extract, evaluate, prefilter and recombine features to generate good feature subsets. Features are evaluated using the learning set. Quadratic features are generated from the initial feature set and added to the entire set of available features. Features are regularized and evaluated with respect to each single features recognition quality. Features are divided into 3 feature subsets F^+,F^- and F^0. Features which are too similar to each other are discarded using a threshold τ. The GA uses the reduced feature sets to evaluate good feature combination called individuals.

centages, also bad behaving feature sets are proposed. It is not always obvious to see, which individuals are finally performing well in the true prediction. To evaluate a general strategy filtering out bad from good individuals, true prediction data can be used. Normally this information is inaccessible. Thus, it cannot be used to judge about the individual, but to understand what filter criterion can be used.
The final generation undergoes a special treatment, which should simplify the judgment over the quality of different individuals. Each individual is scored with 20 different molecule distributions of four quarters. Each quarter can become test prediction set while the remaining 3 quarters become test learning set. This will lead to $20 \cdot 4$ separate test predictions and recalls scored with the scoring function. The resulting MCC values of average test prediction MCC, variance and minimum MCC are computed. Furthermore the average MCC of test learning sets is calculated. As novelty for each individual, the MCC value for the recall of the complete learning set, is computed. These values can be used as indicator for the quality of individuals.

quantity	**description**
MCC(aL)	mean value of all test learning MCCs
MCC(aT)	mean value of all test prediction MCCs
var(MCC(aT))	variance of all test prediction MCCs
min(MCC(aT))	minimum of all test prediction MCCs
MCC(tL)	MCC of complete learning set (recall)
MCC(tP)	MCC of true prediction [5]

Table 2.4: Quality indicators derived from MCC values

The individuals of the final generation are ranked in a table listing all MCC indicators shown in table 2.4. The individuals are ranked according to the MCC(aT) or min(MCC(aT)) values by default. Together with the relation of some other indicators, the minimum MCC of the test prediction is a good measure for the quality of the individuals.

2.3.13 Similarity of learning and prediction data sets

How similar are the learning data sets to the data sets for prediction in the eyes of the features used? The answer to this question can reveal correlations between the different data sets with respect to the features selected for their classification. Furthermore it can indicate how appropriate the selected feature are to classify these data. To calculate the similarity between two molecules i and j of the two data sets x and y, the scalar product is calculated

$$S = \vec{f}_i^{\,set(x)} \cdot \vec{f}_j^{\,set(y)}.$$

To obtain the similarity between two complete sets the normalized sum of scalar products is used:

$$S_{set(x),set(y)} = \frac{1}{N^x\, N^y} \sum_{i,j} \vec{f}_i^{\,set(x)} \cdot \vec{f}_j^{\,set(y)}, \tag{2.43}$$

[5] This information cannot be used in realistic prediction scenarios

where N^x, N^y are the numbers of molecules in the appropriate data sets. To discriminate further, the learning set is divided into a subset of binding and a subset of non-binding molecules. In a real prediction scenario the classes of the molecules in the prediction set are not known. For a broader understanding of the CoEPrA data, assignments of the prediction classes can be used in similarity calculations. This information should not be used to tune the classification. For similarity calculations sequence vectors or feature vectors can be used (see eqn. 2.3, p. 13).

2.4 Alternative methods

In this section some alternative methods to classify molecules shall be briefly summarized. These methods can be used instead of the least square optimization used in the present work.

2.4.1 Hidden Markov Model

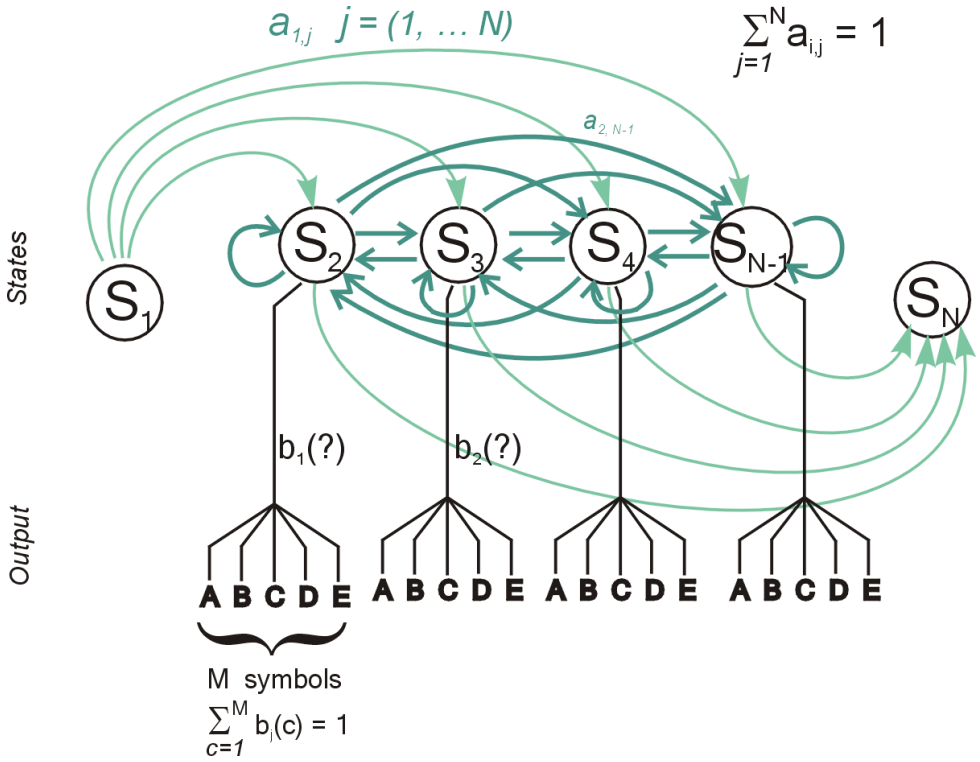


Figure 2.11: Hidden Markov model as a statistical process with a number of different states S. Transition probabilities between state i and j are given by $a_{i,j}$. All transitions of a given state i sum up to a probability of 1. Output probabilities for a given set of symbols (i.e. aa representations) are given by $b_{j,m}$. Output probabilities for all M symbols of each state j sum up to 1. The HMM model H is yielding for different possible output sequences O^k different probabilities $P\left(O^k|H\right)$

Hidden Markov Models (HMM) are statistical models described by a markov process, where the states of the system are unobservable (hidden). The HMM is described by a number of states $S_1, \ldots, S_N$. The first state S_1 is the initial state of the system, while the last state S_N is the terminal state of the model. Each other state generates one output symbol $y_k = m$ each time

it is selected. The timestep t is indicating the current position in the sequence of states of a described path through the model. The states are connected with each other. The transition probability $a_{i,j}$ defines the probability to get from a state i to another state j leading to a $N \times N$ matrix $A = \{a_{i,j}\}$ of transition probabilities. Each state j has a defined probability $b_j(c)$ for generating a certain output symbol $\{c_m\} = \{c_1, \ldots, c_M\}$ out of $m = 1, \ldots, M$ possible symbols. These symbols can encode amino acids in case of peptide sequence predictions. The output probabilities are stored in a $N \times M$ matrix of $B = \{b_{j,m}\}$.
Different output sequences $O = O_1, O_2, \ldots, O_T$, which correspond to peptide sequences in the present classification problem, are generated for the model $H = (A, B)$ with the probability $P(O|H)$. The output sequence O is generated in the model by following a certain path Q of states $q_t = S_i$ for the different time steps t yielding $Q = q_1, q_2, \ldots, q_T$. The probability $P(O|H)$ that a given sequence O is generated is computed in the forward and backward algorithm [45].
The transition probabilities a_{ij} and the output probabilities $b_{j,m}$ are parameters of the system, which are optimized during the training of the model. For optimization an iterative steepest descent algorithm is used to maximize the probability to generate an output sequence of the binding set [45].
A well trained HMM can be used to generate strong binding peptides with a high probability P. Furthermore the HMM is flexible to classify binding peptides of different sequence lengths. Even if anchor positions of binding peptides of a multiple length vary, the HMM can be used for an alignment of the peptides. In the studies of the present work a version of HMM was tested, which has been optimized with respect to the peptide classification problem [46, 47]. The modeled HMM system did not converge well for learning, such that no reasonable result could be computed. Therefore, no results for the HMM are presented in this work.

2.4.2 Random Forest

The random forest approach [48, 49] uses a number of uncorrelated decision trees also referred as classification trees. Each decision tree can independently classify entries from the prediction set and allow therefore the parallelization of the prediction process. A decision tree is a graph model of decisions containing a number of nodes, which are connected by branches. In decision trees each node splits a branch into new branches. The nodes of the tree evaluate certain features m as part of the total number of available features M and decide in dependence of the value of m, which branch to choose to proceed. The terminal state of each path through the tree is represented by a leaf. In case of a classification, each leaf contains just a single class value y, which is the target value connected to the feature values used on the path through the tree. During the learning process the learning set is split in a recursive manner into smaller subsets depending on single feature values. There are different algorithms known to build a decision tree. The Classification and regression trees (CART) method splits off single groups, which are chosen to be as large as possible. Recursion stops when no further gain is made. Another choice for building the trees is the entropy or information gain algorithm. For both algorithms the sum of probabilities of all possible (feature) values j at a given node i has to be calculated.
Each tree is trained N times by a different training set obtained from the random choice of items out of the entire learning set. Remaining items from the learning set are used for test prediction, each time. The test prediction results are used to estimate the error of the tree (bootstrapping procedure).
The advantages of the random forest method are the fast training and the degree of parallelization. It can also handle unbalanced data sets and can be used for both, classification and regression. Dissadvantage is the tendency to overfitting.

Chapter 3

Results

3.1 Structural analysis of peptide bound HLA-A0201 complexes with and without TCR

3.1.1 Superposition of A0201 binding pockets of different crystal structures

If the available crystal structures are divided (see table 2.2 in section 2.2.2 on p.11) into structures cocrystallized with or without TCR, the resulting pattern of superimposed peptides deviates significantly with respect to the different residue positions. The presence of the TCR seems to influence the alignment of the peptide in the MHC binding groove.
To obtain the overlap of the HLA-A0201 binding pockets, the HLA α-heavy chain residues from 0 to 274 are superimposed with the Kabsch algorithm [33][34]. Only protein backbone atoms are considered for the structure alignment. For both groups of MHC crystal structures, with and without cocrystallized TCR, a reference structure is selected to which all remaining structures are aligned to. Thus all RMSD values are 0 for the reference structure. The following tables show the atom-atom RMS deviations with respect to the superimposed α chain of HLA considering only protein backbone atoms. Ideally deviations to the reference structure should be zero, but in reality deviations from the reference positions of the atoms occur . The resolutions of the crystallographic settings and the average B-factors of the HLA α-chains (all atoms of residues 0 - 254) for the different crystal structures are shown in the tables 3.1 and 3.2. The B-factors are temperature factors, which describe possible fluctuations of the atom positions given in the crystal structures.

overview of structures without TCR

pdb ID	**peptide sequence**	**comment**	**RMSD [Å] of HLA α to α**	**B-factor avg. of α**	**resolution [Å]**
1AKJ	ILKEPVHGV	reference structure	0.00	23.02	2.6
1HHG	TLTSCNTSV		1.11	25.50	2.6
1HHI	GILGFVFTL		1.10	22.02	2.5
1HHJ	ILKEPVHGV		1.21	16.19	2.5
1QEW	FLWGPRALV		0.90	17.53	2.2
1DUZ	LLFGYPVYV		1.24	23.76	1.8

Table 3.1: Deviations in HLA α-chains of arranged pMHC structures with bound nona-peptides

overview of structures including TCR

pdb ID	peptide sequence	comment	RMSD [Å] of HLA α to α	B-factor avg.of α	resolution [Å]
1AO7	LLFGYPVYV	reference structure	0.00	42.13	2.6
1LP9	ALWGFFPVL		0.92	15.63	2.0
1QSE	LLFGYPRYV		0.54	60.23	2.8
1QSF	LLFGYPVAV		0.51	56.55	2.8
1BD2	LLFGYPVYV	different TCR	0.95	38.06	2.5

Table 3.2: Deviations in HLA α-chains of arranged pMHC/TCR complexes with bound nona-peptides

3.1.1a Structures without TCR

In the following structural deviations of the ligand peptides in the binding pocket are considered. For each residue position of the ligand the RMSD is calculated for backbone atoms with respect to the selected reference structure.

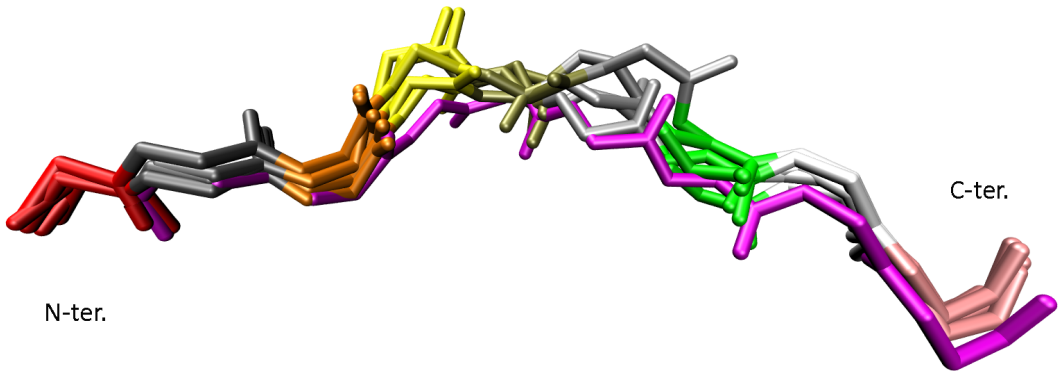
residue	deviations in [Å]						Δ
position	1AKJ[1]	1HHG	1HHI	1HHJ	1QEW	1DUZ	avg.
1	0.00	0.72	0.55	0.16	0.57	0.32	0.46
[2]	0.00	0.85	0.48	0.34	0.20	0.41	0.46
3	0.00	1.23	1.13	0.49	0.29	0.67	0.76
4	0.00	2.06	2.82	0.85	2.51	2.63	2.17
5	0.00	2.43	1.31	0.69	0.78	1.24	1.29
6	0.00	2.22	1.42	0.48	1.66	2.17	1.59
7	0.00	1.37	1.37	0.81	0.67	1.38	1.12
8	0.00	1.14	1.28	1.10	1.25	1.29	1.21
[9]	0.00	1.20	1.49	1.10	0.84	1.30	1.19
Δ1-9	0.00	1.47	1.32	0.67	0.97	1.27	1.14

Table 3.3: Peptide residue specific RMSD of superimposed HLA from pMHC structures. The two HLA-A0201 key residue positions are highlighted with rectangular brackets around the residue position number 2 and 9.

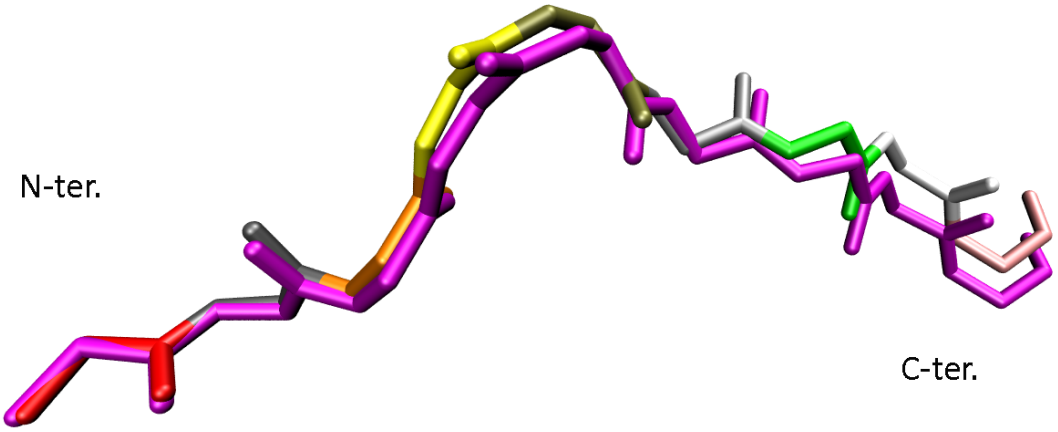
It is easy to recognize that the the N-terminal position of the peptides matches best, since deviations between the structures are low. In most cases the central region, especially the residue position number 4, shows strong deviations for the structures. Towards the C-terminal positions

[1] reference structure

the deviations between the structures are decreasing again. One exception from this rule is structure 1HHJ, which has its' highest deviations for the C-terminal residues. Interestingly, for this structure the ligand peptide is identical to the peptide of the reference structure 1AKJ. Both structures, 1AKJ [28] and 1HHJ [30], contain identical molecules crystallized by different research groups. The deviations in the N-terminal region and the central region of 1HHJ to 1AKJ are the lowest of all compared structures.



(a) all peptides



(b) equal peptides

Figure 3.1: Peptide backbones of superimposed pMHC structures. The N-terminal position of peptides is located left. **a)** all peptides in relation to reference 1AKJ (magenta) **b)** equal ligand peptides between 1HHJ and reference 1AKJ (magenta)

Figure 3.1 is visualizes the stacked peptide backbones of the different superimposed MHC structures. The central region can be identified by the stronger deviation between the peptides. Due to different bond- and torsion angles the structures of the two peptides with the same sequences (1HHJ and reference 1AKJ), deviate more at the C-terminal end. Residues located close to the N-terminal position of the peptides show the best agreement in their backbone positions.

3.1.1b Structures cocrystallized with TCR

A bound T cell receptor is influencing the structure of the ligand peptide in the MHC binding pocket directly. Most peptide residues are exposed to the surface of the pMHC complex, which then can be accessed easily by the adjacent TCR molecule. Interaction between ligand and TCR chains must have an impact to the petides' position in the MHC binding groove. The residue specific position deviations between the aligned peptides are shown in the following table 3.4, which lists only those crystal structures possessing a cocrystallized TCR.

residue	**deviations in [Å]**					**Δ**
position	**1AO7**[2]	**1LP9**	**1QSE**	**1QSF**	**1BD2**	**avg.**
1	0.00	0.27	0.57	0.25	0.23	0.33
[2]	0.00	0.24	0.39	0.24	0.30	0.29
3	0.00	0.34	0.32	0.20	0.40	0.32
4	0.00	0.63	0.50	0.36	0.78	0.57
5	0.00	1.51	0.53	0.48	0.54	0.77
6	0.00	1.09	0.47	0.27	0.49	0.58
7	0.00	1.27	0.45	0.46	0.34	0.63
8	0.00	1.54	0.60	0.66	0.35	0.79
[9]	0.00	1.49	0.22	0.70	0.30	0.68
Δ1-9	0.00	0.93	0.45	0.40	0.41	0.55

Table 3.4: Peptide residue specific RMSD of superimposed HLA from pMHC/TCR complexes. The two HLA-A0201 key residue positions are highlighted with rectangular brackets around the residue position number 2 and 9.

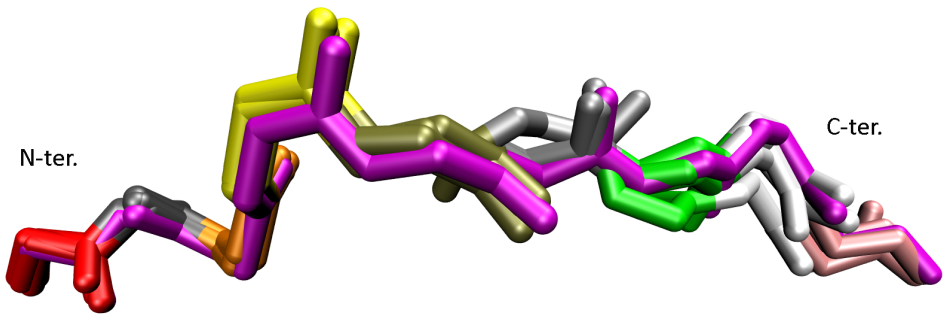


Figure 3.2: Peptide backbones of superimposed pMHC structures with TCR present. The N-terminal position of peptides is located left. All peptides shown in relation to reference 1AO7 (magenta)

[2]reference structure

In this group of MHC structures exists again one structure (1BD2) whose ligand is identical to the ligand of the selected reference structure 1AO7. Here the cocrystallized TCR types are differing. The RMSD values obtained for peptide-peptide deviations are lower than the values obtained from the MHC structures without TCR. This may be explained by the higher peptide ligand homology of these structures containing the TCR. This idea is supported by the fact, that the ligand structure much different from the other ligands of the set, 1LP9, deviates much stronger from the reference structure.
The deviations in the central region of the peptides from the pMHC/TCR structures are marginally different to deviations at the N- or C-terminal ends. The influence of the TCR molecule sitting on top of the binding groove, is decreasing the degrees of freedom that the peptide usually possesses, if no TCR is present. One exception from this observation is again structure 1LP9 containing the highly differing ligand peptide. Here, the RMS deviations from the central part to the C-terminal end are significantly higher than deviations at the N-terminal end of the peptide.

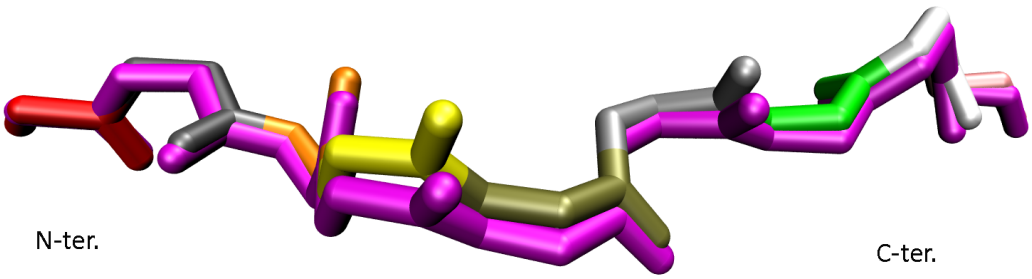


Figure 3.3: Peptide backbones of superimposed structures 1AO7 and 1BD2 with TCR present. The N-terminal position of peptides is located left. The ligand peptides of 1BD2 and reference 1AO7 (magenta) possess the same sequence.

Although the similarity of the backbone structures is higher for peptides cocrystallized with TCR, it is not right to conclude that this effect is caused by the presence of the TCR. Rather the high sequence homology of peptides in the structures with TCR present is responsible for the strong overall agreement of the peptide backbone structures. Nevertheless, sequence homology alone cannot explain the similarity in the peptide structures. In both groups of MHC structures, (with and without TCR), MHC structures with identical peptide sequence show some dissimilarity, which is of the same magnitude as those for structures with different peptide sequences. In case of bare MHC/peptide structures, the peptide structure with the same sequence as the reference structure shows significant deviations in the backbone, which increases towards the C-terminal end of the peptide.

3.1.1c How do deca-peptides align ?

Three HLA-A0201 crystal structures without TCR contain peptides of 10 residues length. It is interesting to see, how they align in the binding pocket of the MHC with their additional residue position. Several possibilities are imaginable:

- The residue number ten sticks out of the C-terminal end of the MHC binding pocket. This can be observed for the structure of 2CLR, where the Glycin is bent out of the pocket of the MHC.

The case, where the first residue at the N-position sticks out of the binding pocket was not observed. This seems to be unlikely, because the atom-atom interactions at the amide end of the peptide chain seems to be very conserved for all binding ligands.

- Central residues of the peptide loop out of the MHC binding pocket or the central residues bend within the binding pocket, compared to the more stretched conformation of binding nona-peptides. This can be observed for the structures 1HHH and 1I4F.

Because there is no unique way how deca-peptides align in the HLA binding pocket, it is not useful to calculate residue position specific RMSD values like it was done in the previous examples. The figures 3.4, 3.5 and 3.6 illustrate the three different peptide structures of deca-peptide ligands obtained by superimposing HLA α-chains to 1AKJ as reference structure. Again with the Kabsch algorithm all residues of the α-chain between residue number 0 to 254 are aligned. Table 3.5 shows average deviations of the backbone atoms of the specified residues from HLA α-chains.

overview of structures without TCR

pdb ID	**peptide sequence**	**comment**	**RMSD [Å] of HLA α to α**	**B-factor avg.of α**	**resolution [Å]**
1AKJ	*nonapeptide*	reference structure			
	ILKEPVHGV	from table 3.2	0.00	23.02	2.6
2CLR	MLLSVPLLLG		0.92	26.62	2.0
1HHH	FLPSDFFPSV		0.54	6.45	3.0
1I4F	GVYDGREHYV		0.51	17.57	1.4

Table 3.5: Deviations in HLA α-chains of arranged pMHC complexes with bound deca-peptides

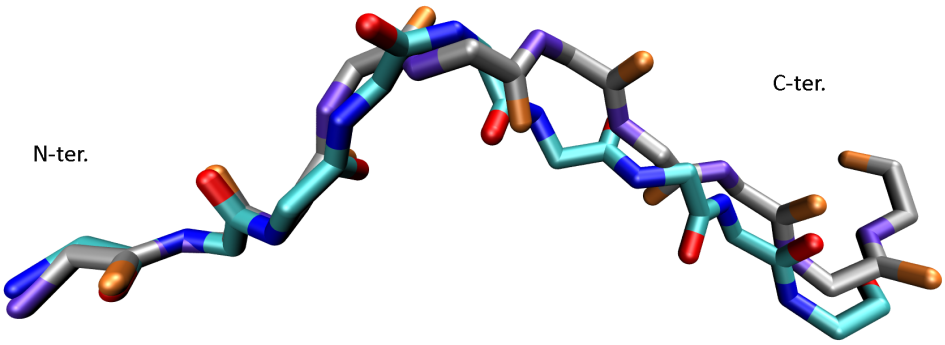


Figure 3.4: Deca-peptide from 2CLR (in silver gray) after superposition of HLA chains with nonapeptide from 1AKJ

As illustrated in figure 3.4, the last residue, a Glycin, sticks out from the superimposed reference nona-peptide. The other two structures, as shown in figures 3.5 and 3.6 have flexible loops in the central region of the peptide with respect to the reference peptide structure. The N-terminal residues are very conserved in all structures and show low variance.

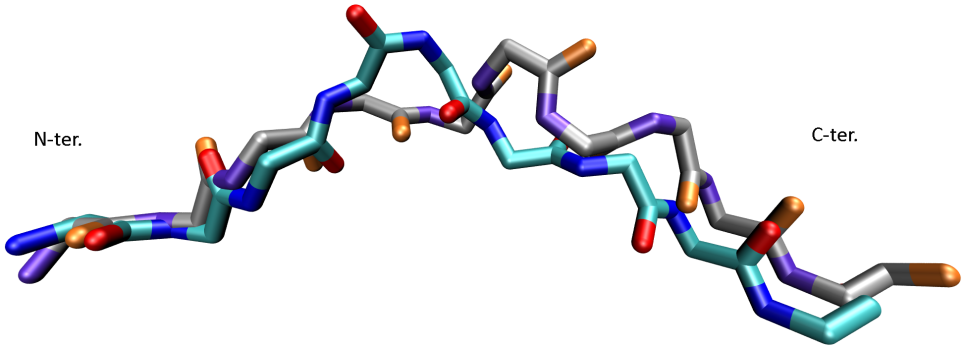


Figure 3.5: Deca-peptide from 1HHH (in silver gray) after superposition of HLA chains with nonapeptide from 1AKJ

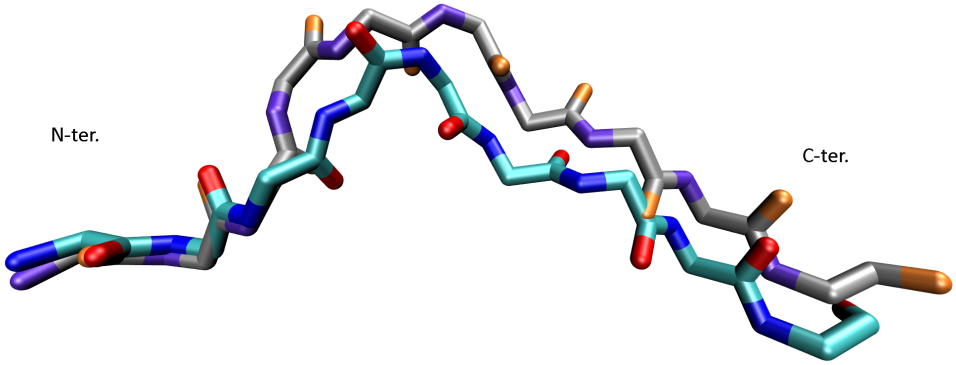


Figure 3.6: Deca-peptide from 1I4F (in silver gray) after superposition of HLA chains with nonapeptide from 1AKJ

3.1.2 Intermolecular contact distances

In the following sections atom-atom contact distances between the peptide ligand and one of the chains from either MHC protein HLA-A0201 or TCR protein are examined. A contact is defined by those atom pairs whose distances are less or equal than a defined threshold distance. For hydrophilic and hydrophobic interactions the contact distance threshold is chosen as 3.5 Å. The ligand residue positions for the atom pair contacts are again analyzed separately.

3.1.2a Crystal structures without TCR

The structures 1AKJ, 1HHG, 1HHI, 1HHJ, 1QEW and 1DUZ contain just the ligand peptide bound to HLA-A0201. The tables 3.6 - 3.11 list contacts counted for each residue position of the ligand peptide to the HLA α-chain. Backbone and side chain atoms are counted separately and the table discriminates between hydrophilic and hydrophobic atom-atom contacts. The table 3.12 gives the average contacts calculated over all 6 structures.

While hydrophobic atom-atom interactions are mainly a consequence of a cluster of bulky uncharged groups, hydrophilic atom-atom interactions result in formation of hydrogen bonds or salt bridges between side chains, which are specific interactions for a selected number of functional groups. Nevertheless, these interactions can be residue type unspecific if they occur with peptide backbone atoms.

1AKJ			ILE	LEU	LYS	GLU	PRO	VAL	HIS	GLY	VAL
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	2	-	-	-	-	1	3
		hydrophobic	6	-	-	-	-	-	-	2	2
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	1	-	-	1	-	-
		hydrophobic	3	2	1	2	-	3	-	-	3

Table 3.6: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1AKJ

1QEW			PHE	LEU	TRP	GLY	PRO	ARG	ALA	LEU	VAL
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	2	-	-	-	-	1	4
		hydrophobic	-	-	-	-	-	-	-	-	2
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	1	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	7	4	-	-	1	1	-	-	5

Table 3.7: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1QEW

1HHI			GLY	ILE	LEU	GLY	PHE	VAL	PHE	THR	LEU
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	-	-	-	-	-	1	2
		hydrophobic	8	1	1	-	-	-	1	1	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	1	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	1	-
		hydrophobic	-	6	1	-	1	2	3	1	3

Table 3.8: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1HHI

1HHJ			ILE	LEU	LYS	GLU	PRO	VAL	HIS	GLY	VAL
peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	5
		hydrophobic	4	-	1	-	-	-	-	1	4
side chain	backbone	hydrophilic	-	-	1	-	-	-	-	-	-
		hydrophobic	-	1	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	1	-	-
		hydrophobic	4	4	1	3	2	2	-	-	1

Table 3.9: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1HHJ

1HHG			THR	LEU	THR	SER	CYS	ASN	THR	SER	VAL
peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	5
		hydrophobic	6	3	1	-	-	-	1	2	4
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	2	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	1	-
		hydrophobic	1	3	1	-	-	-	-	2	-

Table 3.10: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1HHG

1DUZ			LEU	LEU	PHE	GLY	TYR	PRO	VAL	TYR	VAL
peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	5
		hydrophobic	3	-	2	-	-	-	1	-	3
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	3	3	1	-	-	-	-	-	-

Table 3.11: Atom-atom contacts between peptide ligand and HLA-A0201 molecule for 1DUZ

average pMHC pattern for A0201

peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	4
		hydrophobic	4	1	1	-	-	-	-	1	2
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	1	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	3	4	1	1	1	1	-	-	2

Table 3.12: Atom-atom contacts between peptide ligand and HLA-A0201 molecule average pattern

Very conserved atom-atom interactions occur at the N- and C- terminal positions of the peptide backbone. Namely the amide- and carboxy-group of the peptide endgroups interact with residues of the HLA-A0201 protein very intensively. Side chain - side chain interactions between peptide and HLA-A0201 are depending significantly of the peptide sequence and therefor varying for the examined examples. Around residue position 2 of the peptide, side chain - side chain interactions to are mostly conserved. This is reflecting the key residue position 2, where Leucin is the preferred amino acid. With the exception of side chain - side chain interactions, central residue positions barely have atom-atom contacts with HLA-A0201 residues.

3.1.2b Crystal structures with TCR present

Here, atom-atom interactions of structures containing TCR together with the HLA-A0201 molecule and the bound ligand peptide are analyzed. Those structures are 1AO7, 1BD2, 1QSF, 1QSE and 1LP9. Most structures are crystallized with human A6 type TCR, while two structures (1BD2 with A7 and 1LP9 with AHIII 12.2) contain a different TCR type. The tables 3.13 - 3.17 list interactions between peptides and MHC HLA-A0201 molecules as well as between peptides and TCR molecules. In the table 3.18 average contacts of all five crystal structures containing TCR are shown.

Almost all considered structures with TCR, contain a ligand peptide, whose sequence differs just in one residue position. Only structure 1LP9, which possesses a different TCR type than the four other structures, contains a ligand peptide with a significant different sequence. The atom-atom interaction pattern is therefore similar for all considered structures with TCR present. Most of the atom-atom interactions between peptide and HLA-A0201 residues, are found in residues 1-3 and 7-9. Like for peptide structures without TCR, peptide backbone atoms have conserved interactions with HLA-A0201 residues especially for N- and C- terminal residue positions. Central residues are not affected, regarding the interactions between the peptide and the HLA-A0201 protein.
In contrast to the HLA-A0201 protein, central peptide residues are in contact to TCR residues, especially for the side chain to side chain interactions of residue 5. There are backbone-backbone contacts between peptide chain and the TCR protein for peptide residue position 4[50]. Differences in the interaction pattern can be found for different TCR types and differences arise also from variation in the peptide sequence.

1AO7	TCR type: A6 human		LEU	LEU	PHE	GLY	TYR	PRO	VAL	TYR	VAL
peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	3
		hydrophobic	2	-	1	-	-	-	-	1	1
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	3	3	1	-	-	-	-	-	-
	TCR										
backbone	backbone	hydrophilic	-	-	-	1	-	-	-	1	-
		hydrophobic	-	-	-	1	-	-	1	-	-
backbone	side chain	hydrophilic	-	1	-	1	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	2	-	-	-	-
		hydrophobic	-	-	-	-	2	-	-	1	-

Table 3.13: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for 1AO7

1BD2	TCR type: A7 human		LEU	LEU	PHE	GLY	TYR	PRO	VAL	TYR	VAL
peptide	protein	type	1	2	3	4	5	6	7	8	9
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	4	3	1	-	-	-	-	1	4
		hydrophobic	5	-	1	-	-	-	1	1	2
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	1	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	3	2	2	-	-	-	-	4	4
	TCR										
backbone	backbone	hydrophilic	-	-	-	1	-	-	-	1	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	1	3	-
side chain	side chain	hydrophilic	-	-	-	-	1	-	-	-	-
		hydrophobic	1	-	-	-	4	-	-	-	-

Table 3.14: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for 1BD2

1QSF	TCR type: A6 human		LEU	LEU	PHE	GLY	TYR	PRO	VAL	ALA	VAL
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	3
		hydrophobic	3	3	2	-	-	-	1	4	2
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	7	3	4	-	-	-	-	1	3
	TCR										
backbone	backbone	hydrophilic	-	-	-	1	-	-	-	-	-
		hydrophobic	-	-	-	1	-	-	-	-	-
backbone	side chain	hydrophilic	-	1	-	-	-	-	-	-	-
		hydrophobic	-	-	-	1	-	-	-	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	1	-	-	-	-
		hydrophobic	-	-	-	-	3	-	-	-	-

Table 3.15: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for 1QSF

1QSE	TCR type: A6 human		LEU	LEU	PHE	GLY	TYR	PRO	ARG	TYR	VAL
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	4
		hydrophobic	6	5	2	-	-	-	1	3	5
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	2	-	-	-	-	-	1	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	8	3	6	-	-	3	-	-	2
	TCR										
backbone	backbone	hydrophilic	-	-	-	1	-	-	-	-	-
		hydrophobic	-	-	-	2	-	-	-	-	-
backbone	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	3	-	-
		hydrophobic	-	-	-	-	-	-	2	1	-
side chain	side chain	hydrophilic	-	-	-	-	2	-	-	-	-
		hydrophobic	-	-	-	-	5	-	-	3	-

Table 3.16: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for 1QSE

1LP9	TCR type: AHIII 12.2		ALA	LEU	TRP	GLY	PHE	PHE	PRO	VAL	LEU
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	1
backbone	side chain	hydrophilic	3	2	2	-	-	-	-	1	4
		hydrophobic	5	-	1	-	-	-	1	-	4
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	1	3	1	-	-	-	-	-	-
	TCR										
backbone	backbone	hydrophilic	-	-	-	2	1	-	-	-	-
		hydrophobic	-	-	-	3	-	-	-	-	-
backbone	side chain	hydrophilic	-	-	-	1	-	-	-	-	-
		hydrophobic	-	-	-	-	-	1	1	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	1	-	1	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	1	-

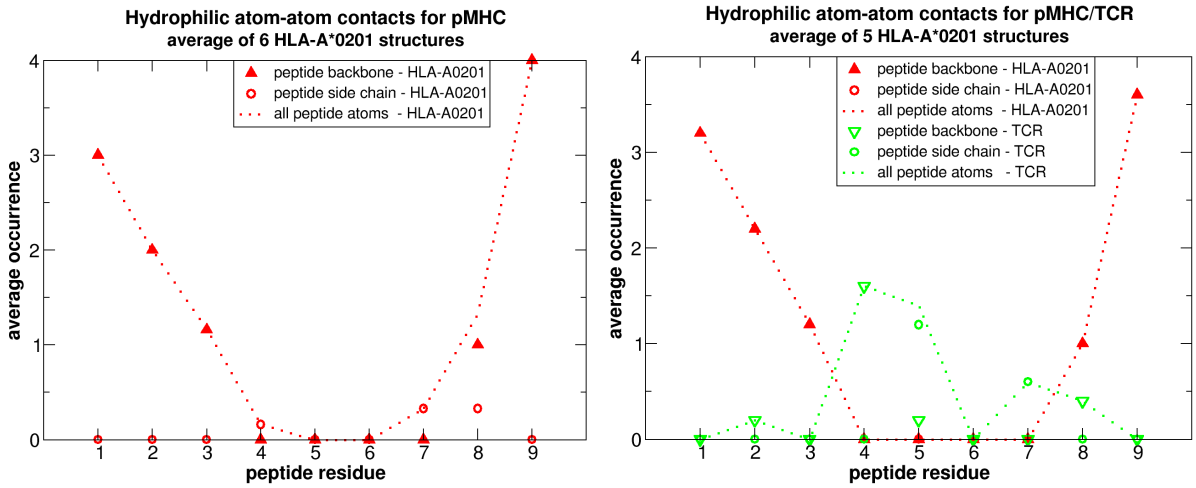
Table 3.17: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for 1LP9

average pMHC/TCR pattern for A0201										
peptide	**protein**	**type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	HLA-A0201										
backbone	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
backbone	side chain	hydrophilic	3	2	1	-	-	-	-	1	4
		hydrophobic	4	2	1	-	-	-	1	2	3
side chain	backbone	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	4	3	3	-	-	1	-	1	2
	TCR										
backbone	backbone	hydrophilic	-	-	-	1	-	-	-	-	-
		hydrophobic	-	-	-	1	-	-	-	-	-
backbone	side chain	hydrophilic	-	-	-	-	-	-	-	-	-
		hydrophobic	-	-	-	-	-	-	-	-	-
side chain	backbone	hydrophilic	-	-	-	-	-	-	1	-	-
		hydrophobic	-	-	-	-	-	-	1	1	-
side chain	side chain	hydrophilic	-	-	-	-	1	-	-	-	-
		hydrophobic	-	-	-	-	3	-	-	1	-

Table 3.18: Atom-atom contacts between peptide ligand and HLA-A0201 molecule as well as between peptide ligand and TCR for average pMHC/TCR pattern

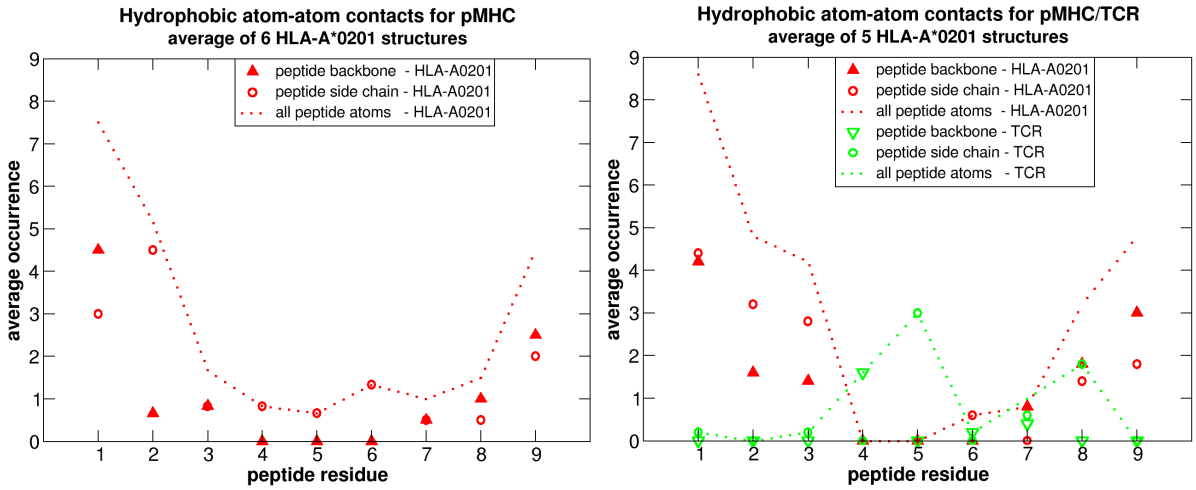
It is hard to predict how larger differences in the ligand peptide sequence will alter the interaction pattern to the TCR chains, since all the examined peptide structures barely differ in their sequence. This might also be affected by the increased selectivity of the complex, if a specific TCR is present[51][10].

3.1.3 Comparison of peptide binding in HLA-A0201 structures with and without TCR



(a) pMHC structures (b) pMHC+TCR structures

Figure 3.7: Summary of **hydrophilic contacts** formed by salt bridges and hydrogen bonds between peptide and protein atoms averaged for **a.)** structures without TCR **b.)** structures including TCR



(a) pMHC structures (b) pMHC+TCR structures

Figure 3.8: Summary of **hydrophobic contacts** between peptide and protein atoms averaged for **a.)** structures without TCR **b.)** structures including TCR

The most remarkable conclusion from the analysis of the HLA crystal structures is that the central region of the bound peptide is not specifically recognized by the HLA-A0201 molecule. In contrast, the TCR interacts especially with the side chains of the central residues. A very tight interaction between N- and C- terminal peptide residues and the HLA-A0201 protein is conserved through all examined structures. The graphs in figure 3.7 highlight these observations. Hydrophilic interactions between HLA-A0201 and ligand peptide are predominantly formed between HLA protein and peptide backbone atoms. These interactions are residue type unspecific compared to hydrophilic interactions with peptide side chain atoms. This goes along with the observation for the examined HLA-A0201 that at several peptide residue positions (pos. 1,2,6,7 and 9, see table A.3 in the appendix section A.0.1) unpolar, hydrophobic amino acids are preferred. More residue type specific are the key positions 2 and 9 (see fig. 3.8), where the hydrophobic amino acids leucine or valine respectively are located with a higher probability (see table A.3). The graphs in figure 3.8 show that hydrophobic interactions occurr between the HLA-A0201 protein and the peptide for all residue positions 1 to 9. Specific for large, bulky, unpolar amino acids are those hydrophobic interactions, caused by peptide side chain contacts. Thus for HLA-A0201 the key positions 2 and 9 of the binding peptide are known to cover bulky, unpolar amino acids, which cause hydrophobic interactions of the peptides' side chain atoms and the protein, as shown in figure 3.8 in the graph (a) for pMHC structures as well as in graph (b) for pMHC + TCR structures. The side chain interactions with HLA are marked by the red open circles in figures 3.7a/b and 3.8a/b. For position 2 in particular, this number is higher than the appropriate number of hydrophobic interactions to peptide backbone atoms marked by the red triangles (see fig. 3.8a/b). In presence of the TCR, hydrophobic interactions with the central residues are formed almost exclusively with the TCR protein. The HLA-A0201 protein is almost not interacting with peptide residues 4,5 and 6 when the TCR is present. For structures with TCR absent, some hydrophobic interactions of HLA-A0201 protein and peptide side chain atoms on position 4,5 and 6 can be observed. Possibly the presence of TCR is supressing those HLA-A0201 - peptide interactions by pulling the peptide chain out of the MHC binding groove. However this observation may also be an artefact due to the small sequence variations of the examined crystal structures including the TCR.

3.1.4 Conserved contact residues of the HLA-A0201 α-chain

The tables 3.19 and 3.20 highlight those residues of the α-chain of A0201, which form hydrophilic interactions with the ligand peptide for structures without and with TCR present. Table 3.19 lists only the pMHC structures, where the interactions of 8 residues are conserved and found in all 6 examined structures. HLA-A0201 residues, which interact with side chain atoms of the peptide, are marked in the table by the $\oplus$symbol. In the case of glutamine residue 155, interactions are only found to peptide side chains. In both cases of occurrences the peptide interaction partner was a histidine on residue position 7 of the ligand peptide.

The table 3.20 lists hydrophilic interacting residues from HLA-A0201 with peptide atoms of crystal structures with TCR present. Due to the restricted variability of the peptide ligands' sequence, only minor differences can be observed. There are in no case hydrophilic interactions of HLA with peptide side chains occurring. Thus, all the hydrophilic interactions observed here are targeting the peptide backbone atoms.

amino acid	**Tyr**	**Glu**	**Arg**	**Lys**	**His**	**Asp**	**Tyr**	**Tyr**	**Thr**	**Lys**	**Trp**	**Gln**	**Tyr**	**Tyr**
res. no.	**7**	**63**	**65**	**66**	**70**	**77**	**84**	**99**	**143**	**146**	**147**	**155**	**159**	**171**
atom label	OH	OE1	NH2	NZ	NE2	OD1	OH	OH	OG1	NZ	NE1	OE1	OH	OH
structures														
1AKJ	+	+	⊕	+	+	+	+	+	+		+	⊕	+	+
1QEW	+	+		+	+	+	+	+	+	+	+		+	+
1HHG	+	+		+		+⊕	+	+	+	+	+		+	+
1HHI	+	+		+		+			+	+	+		+	+
1HHJ	+	+		+		+	+	+	+	+	+	⊕	+	+
1DUZ	+	+		+		+	+	+	+	+	+		+	+
Σ	6	6	1	6	2	6	5	5	6	5	6	2	6	6

Table 3.19: Hydrophilic interactions between peptide and HLA-A0201 atoms in 6 pMHC structures formed by conserved A0201 residues. Interactions with peptide residue side chains are marked by ⊕.

amino acid	**Tyr**	**Glu**	**Lys**	**His**	**Asp**	**Tyr**	**Tyr**	**Thr**	**Lys**	**Trp**	**Tyr**	**Tyr**
res. no.	**7**	**63**	**66**	**70**	**77**	**84**	**99**	**143**	**146**	**147**	**159**	**171**
atom label	OH	OE1	NZ	NE2	OD1	OH	OH	OG1	NZ	NE1	OH	OH
structures												
1AO7	+	+	+		+	+	+	+		+	+	+
1BD2	+	+	+		+	+	+	+	+	+	+	+
1QSE	+	+	+		+	+	+	+	+	+	+	+
1QSF	+	+	+		+	+	+	+		+	+	+
1LP9	+	+	+	+	+	+	+	+	+	+	+	+
Σ	5	5	5	1	5	5	5	5	3	5	5	5

Table 3.20: Hydrophilic interactions between peptide and HLA-A0201 atoms in 5 pMHC/TCR structures formed by conserved A0201 residues

3.1.5 Discussion and summary of section 3.1

- **TCR limits ligand peptide flexibility:** The peptide bound to HLA-A0201 has more freedom to align in the MHC binding pocket in absence of TCR.

- **Hydrophilic HLA-A0201-peptide interactions:** Hydrophilic interactions between HLA-A0201 and peptide atoms are mainly formed with peptide backbone amide nitrogen or carbonyl oxygen atoms, which are residue type unspecific interactions.

- **Hydrophobic HLA-A0201-peptide interactions:** Mostly hydrophobic interactions are formed between predominantly hydrophobic residues of the peptide at positions 2,9 and the residues of the HLA-A0201 protein.

- **Interactions of HLA-A0201 to different peptide parts:** Interactions between the HLA-A0201 protein atoms and peptide atoms mainly occur to the N- and C- terminal residues of the peptide. The side chains of the central residues 4,5 and 6 are rarely interacting with the A0201 protein.

- **TCR-peptide interactions:** TCR protein atoms interact with central residues of the peptide. Although there are few interactions with terminal peptide residues, most interactions are found with the residues 4 and 5 of the peptide.
- **Deca-peptides as ligands:** Ligand peptides with a residue length of ten can bind to HLA-A0201 if their central residues loop out of the binding groove. Alternatively, in case of a C-terminal Glycine at residue position 10, this residue is hanging out of the binding groove.
- **HLA-A0201 residues:** Hydrophilic interactions between the HLA-A0201 protein and the peptide backbone are caused by the 8 conserved residues Tyr7, Glu63, Lys66, Asp77, Thr143, Trp147, Tyr159 and Tyr171 from the A0201 α-protein chain.

The results obtained in the analysis of the crystal structures suggest that C- and N- terminus play an important role for the ligand peptide binding to HLA-A0201. This finding is supported by the fact that key residues for HLA-A0201 peptides are located at residue positions 2 and 9. On the other hand, the central residues 4,5 and 6 contribute little to the specific recognition of an A0201 binding peptide. Important is the finding that the examined types of TCR A6 and A7 interact especially with the central peptide residues 4,5 and 6. The present key positions 2 and 9 for the HLA-A0201 - peptide binding are suggesting a tight binding of these peptide positions to the MHC binding groove, such that there is little space for interactions of these peptide positions with the TCR. Hence, to find immuno-active binding peptides to HLA-A0201 it is crucial to consider all peptide positions, since those parts of the peptide, which are not so important for binding to the HLA-A0201 have a high importance for the TCR.
The accuracy of the obtained results is based on this small set of available crystal structures. Especially for the complexes, which contain TCR, the peptide sequence variance is small. Some findings could be an artefact of this limitation.
How general are these findings? In this work crystal structures of HLA-A0201 have been examined. For this allele most crystal structures are publicly available. It was also important that a number of crystal structures exist, which contain an attached TCR. It is difficult to expand the finding made in this section to other HLA types especially, since other types vary in their key positions. For subtypes of the same super family HLA-A02*, which are containing the same key positions, one can assume a similar HLA - peptide interaction pattern. For the different types of TCR this finding is even more difficult. The peptide-MHC specific recognition sequence of different TCR types is highly variant and different TCR types are only recognizing specific HLA types. The small pool of available crystal structures prohibits a broader analysis of different TCR types with respect to the various types of HLA.

3.2 Linear Scoring function and Support vector machines

In this section, results for the scoring function (least square method LSM) using sequence information as descriptors are compared to results from the support vector machine (SVM) provided with the same descriptor input.

3.2.1 An example for deriving parameters of LSM and SVM

Different methods provide different results, if the learning set used to train the algorithms is modified. A typical example is discussed by using half of the binding sequences from $\mathbb{S}^+$ as input. These 269 binding sequences and the same number of non-binding sequences from $\mathbb{S}^-$ are used for learning. For simplicity, every binding sequence of table A.1, which possesses an even index number was selected for the learning set. The 269 non-binding sequences were selected randomly from the 10,000 total available non-binding sequences. The remaining 269 sequences from binding set and another randomly chosen 269 non-binding sequences from the non-binding set $\mathbb{S}^-$ are used for prediction.
The parameters $\vec{w}$ and b are obtained for the scoring function by applying the method of the least square optimization (LSM) as described in section 2.3.2c, eqn.2.6. With knowledge of these parameters the linear equation $f(\vec{x}) = \vec{w}^t \cdot \vec{x} + b$ can be solved. To solve eqn. 2.18 in section 2.3.6 for the support vector machine the parameters $\vec{w}$ and b also need to be calculated. To obtain the parameters, listed in table 3.21, the weighting factor w^+ for the scoring function - according to eqn. 2.10 - was set to $w^+ = 0.45$ and therefore $w^- = 1 - w^+ = 0.55$. The lambda regularization term for the scoring function was set to $\lambda_w = 10^{-6}$.

3.2.2 Recognition and prediction on a prototypical example

After training with the learning data set, the quality of the methods is evaluated for recognition and prediction. For recognition the same learning set is presented to the trained scoring function, while for prediction the prediction data set with unknown target values is examined.
The following results were obtained for recognition and prediction of 538 leaning sequences and 538 prediction sequences:

	recognition results				prediction results			
	accuracy		missclassified		accuracy		missclassified	
method	binding	non-binding	binding	non-binding	binding	non-binding	binding	non-binding
LSM	93.3%	95.9%	18	11	92.9%	86.2%	19	37
SVM	93.3%	94.4%	18	15	92.2%	89.2%	21	29

The support vector machine was selecting 238 support vectors for this classification task, where 115 were derived from binding and 123 from non-binding sequences of the learning set. In recognition both methods missclassify 18 peptides as "False negatives" (FN), which have to be binding peptides. 14 of those 18 false negative classified peptides are identical for both methods. There are 11 missclassified binding peptides ("False positives" (FP)) for the scoring function and 15 FP peptides for the support vector machine. From those false positives 10 are identical for both methods.

amino	position 1		position 2		position 3		position 4		position 5		position 6		position 7		position 8		position 9	
acid	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM	LSM	SVM
ALA	-0.02	0.16	-0.07	0.12	0.14	0.44	-0.21	-0.21	-0.26	-0.15	-0.19	-0.06	0.07	0.26	-0.16	-0.13	-0.10	-0.05
HIS	-0.01	0.11	-0.26	-0.16	0.18	-0.02	-0.41	-0.11	-0.08	0.08	0.08	-0.06	-0.38	-0.10	-0.13	-0.11	0.10	-0.03
GLU	-0.16	-0.24	-0.20	-0.08	-0.37	-0.34	0.02	0.13	-0.13	-0.10	-0.29	-0.27	-0.38	-0.30	-0.04	0.16	-0.21	-0.13
GLN	-0.31	-0.17	-0.28	-0.11	0.01	0.13	-0.20	-0.14	0.25	0.25	-0.14	-0.05	-0.33	-0.14	-0.07	0.10	0.09	-0.02
ASP	-0.39	-0.22	-0.39	-0.33	-0.06	0.05	0.10	0.12	-0.38	-0.36	-0.36	-0.26	-0.41	-0.39	-0.02	-0.15	-0.09	-0.19
ASN	-0.32	-0.20	-0.80	-0.58	-0.04	0.10	-0.29	-0.26	-0.16	0.01	-0.18	-0.11	-0.23	-0.07	-0.51	-0.38	-0.25	-0.11
LEU	-0.20	-0.15	0.78	1.50	-0.10	0.20	-0.24	-0.11	-0.22	-0.10	0.00	-0.03	0.15	0.40	-0.01	0.24	0.40	0.63
GLY	-0.03	0.04	-0.22	-0.16	-0.17	-0.15	0.06	0.25	-0.03	0.32	-0.09	0.05	-0.30	-0.25	-0.01	0.08	-0.29	-0.37
LYS	-0.19	0.02	-0.54	-0.36	-0.16	-0.12	0.08	0.27	-0.16	-0.10	-0.23	-0.26	-0.44	-0.28	-0.14	0.06	-0.42	-0.34
SER	-0.19	-0.15	-0.39	-0.39	-0.03	0.03	-0.04	0.05	-0.24	-0.32	-0.04	0.12	-0.11	0.04	-0.20	-0.05	-0.09	0.01
VAL	-0.05	0.02	0.05	0.04	-0.03	0.13	-0.09	-0.08	0.05	0.30	0.07	0.38	0.02	0.00	-0.12	-0.06	0.48	0.95
ARG	0.19	0.23	-0.56	-0.14	-0.28	-0.03	-0.09	0.07	0.00	0.13	-0.52	-0.35	-0.32	-0.20	-0.21	-0.04	-0.35	-0.27
THR	-0.16	-0.18	0.08	0.23	0.27	0.19	-0.01	-0.01	-0.09	-0.02	-0.07	-0.09	-0.01	0.16	0.01	0.09	0.01	0.24
PRO	-0.04	0.10	-0.28	-0.16	-0.15	-0.13	0.01	0.22	-0.05	0.18	0.05	0.23	0.13	0.17	0.16	0.28	-0.49	-0.32
ILE	0.04	0.15	0.29	0.35	-0.07	-0.14	0.31	0.29	0.08	0.07	0.14	0.53	-0.03	0.11	-0.18	-0.12	0.29	0.53
MET	-0.13	0.15	0.69	0.88	-0.27	-0.01	-0.25	-0.14	-0.33	-0.18	-0.28	-0.03	-0.12	0.00	0.05	0.04	0.20	0.14
PHE	-0.04	0.10	-0.26	-0.16	-0.07	0.08	-0.29	-0.20	-0.20	0.12	0.19	0.25	0.20	0.39	-0.36	-0.08	-0.28	-0.19
TYR	0.11	0.18	-0.44	-0.44	-0.28	-0.34	-0.46	-0.26	-0.24	-0.13	0.08	-0.01	0.06	-0.06	-0.18	0.03	-0.37	-0.37
CYS	-0.30	-0.07	0.54	0.04	-0.58	-0.22	-0.18	0.08	0.15	0.08	-0.09	-0.02	0.17	0.12	0.14	0.02	-0.17	-0.04
TRP	0.19	0.13	0.28	-0.07	0.05	0.16	0.17	0.02	0.05	-0.07	-0.14	0.03	0.26	0.14	-0.01	0.01	-0.46	-0.07

Table 3.21: The 180 parameters for the position dependent amino acid types obtained for the linear scoring function after least square optimization (LSM) and the parameters obtained for linear kernel support vector machines (SVM). For learning 269 binding sequences from $\mathbb{S}^+$ are used (all even numbered peptide sequences from table A.1 in the Appendix) and the same number of non-binding sequences from the 10,000 sequences of $\mathbb{S}^-$ are used. The resulting b parameters are $b_{LSM} = 0.22$ and $b_{SVM} = 1.09$.

Recognition performance for both methods is almost the same with a slight advantage for the LSM. In Prediction both methods differ slightly. Especially the non-binding peptides are classified better with the SVM, although there is a small advantage in prediction of the binding peptides for the LSM. The difference in prediction of non-binding peptides could indicate some learning by heart in case of LSM, since recognition and prediction rate for non-binding peptides is for this method more apart.

3.2.3 Influence of the weighting parameter w^+ and the regularization parameter λ_w for the LSM results

As described in section 2.3.2d by eqn. 2.10 and 2.11 the two parameters for weighting and regularization influence the behavior of the least square optimization of the scoring function. In this section the influence of these parameters to the recognition or prediction results will be demonstrated.
The weighting factors w^+ and w^- discriminate binding from non-binding data of the learning set during the training of the method (least square optimization method). The weighting parameter w^+ has always to be seen in relation with w^-, which is defined by $1-w^+ = w^-$. Both parameters determine the weight given to all molecules of either the binding or the non-binding class during the LSM. If the parameters are chosen to be 0.5 each, both classes are treated with the same weight. The influence of different weights is shown for the recognition of the scoring function for a learning data set composed of an equal number of binding and non-binding peptides. For the following case 200 binding and 200 non-binding peptides were selected for the learning set $\mathbb{S}_{learn}$. The peptides from the learning set are ordered ascending by their target values $f(\vec{x}_n) < f(\vec{x}_{n+1})$. Target values are assigned by the scoring function during the recognition after the learning. Thus all values increase monotonously with the sequence number n. In the fig. 3.9 the scoring functions $f(\vec{x}_n)$ are plotted as a function of the index n for different weights of w^+. Peptides with wrong recognized classes are marked by crosses. For comparison a scoring function, which uses linear and quadratic terms (QSM) (see section 2.3.3) is added. To limit the number of free parameters for the QSM, quadratic terms are using amino acid categories. For this example nine categories are used as follows:

1. **hydrophobic** containing `AVM`
2. **histidine** with only `H`
3. **positive charged** with `KR`
4. **negative charged** with `ED`
5. **polar** with `QNSTY`
6. **large** with `LIFW`
7. **glycine** with only `G`
8. **proline** with only `P`
9. **cysteine** with only `C`.

The number of free parameters is given by $N + (M \cdot (M+1))/2$ where N is the number of free parameters of the linear terms and M is the number of amino acid types contributed by the quadratic term. With $N = 180$ and $M = 81$ this leads to 3501 free parameters for the QSM. It is obvious that the QSM scoring function is learning by heart for the recognition of the learning data set.

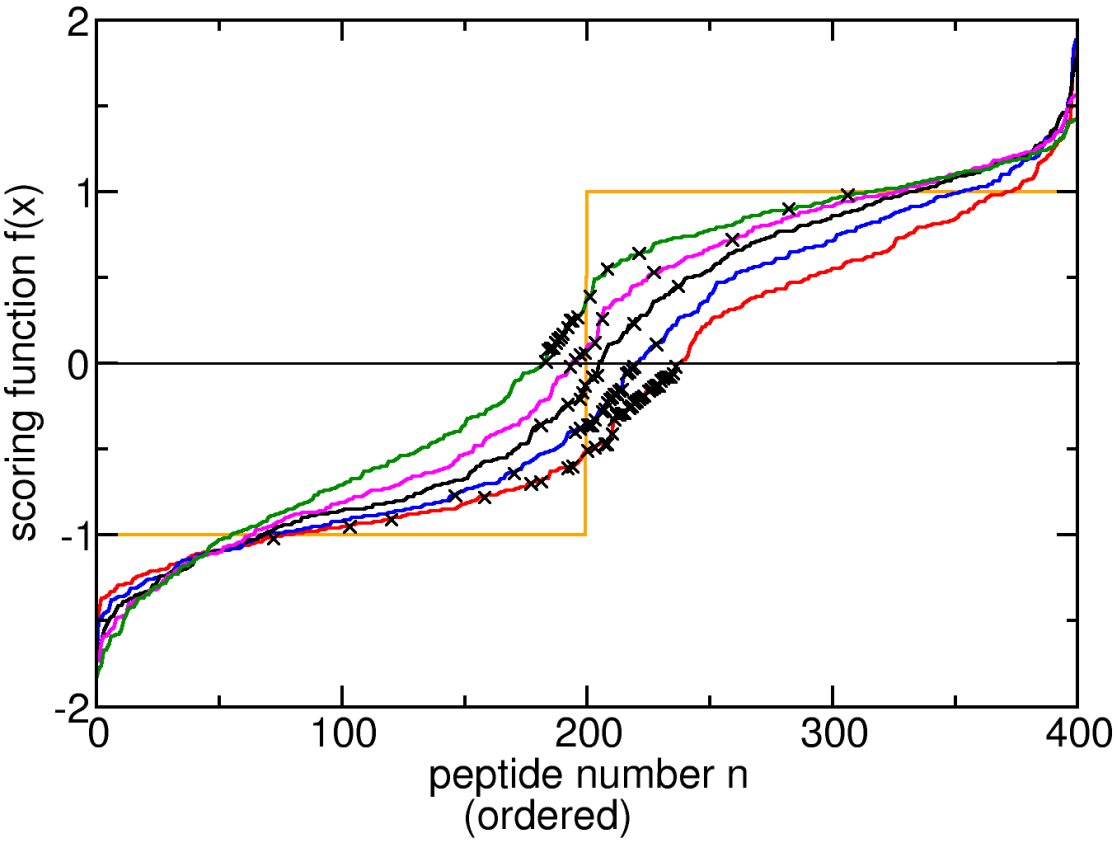


Figure 3.9: Different weights w^+ of binding peptides effect the scoring function $f(\vec{x})$. Recognition of the learning set, containing 200 binding and 200 non-binding peptides, is shown. From the top to the bottom the scoring functions refer to weights w^+ of (green) 0.8, (magenta) 0.6, (black) 0.4, (blue) 0.2 and (red) 0.1. Wrong classified peptides are marked by crosses. The orange line marks the desired step function shape. Here the orange curve is obtained by a scoring function using all linear terms plus quadratic terms of nine categories. The orange QSM plot using a weight $w^+ = 0.5$ shows perfect learning by heart behavior.

weights w^+	**No. Incorrect binding**	**No. Incorrect non binding**
0.8	0	17
0.6	1	7
0.4	7	2
0.2	21	1
0.1	37	0

Table 3.22: Number of missclassified peptides of the scoring function for different weights w^+ as shown in figure 3.9.

A scoring function with optimal behavior should have the shape of a step function, where the first 200 non-binding peptides are classified with a target value of -1 and the 200 binding

peptides on the other side are classified with a target value of +1. The quadratic scoring function in figure 3.9 possesses this behavior. All other curves correspond to results obtained with linear scoring functions using different weights and show a monotonically increasing slope. Here the positive or negative part of the step like functions is shifted, depending on the applied weight w^+. For large weights of w^+ the positive step of the scoring function is very pronounced, while the negative step is less distinct. The opposite is the case for small weights of w^+.
The regularization parameter λ_w can be used to suppress learning by heart. How is this parameter influencing the prediction capability of the scoring function? In the following analysis different learning sets are generated out of the entire set of available binding and non-binding peptides. The weighting factor is always kept constant with $w^+ = 0.35$ and the size of the prediction set is also constant with 150 binding and 150 non-binding peptides. For training three different learning sets are generated, one small set with 50 binding and 50 non-binding peptides, one medium size set of each 100 binding and non-binding peptides and one large set of 350 binding peptides and 1,000 non-binding peptides. The peptides in each learning set are selected by chance from the complete pools of binding and non-binding peptides $\mathbb{S}^+$ and $\mathbb{S}^-$. The peptides for the prediction set are randomly chosen from the remaining peptides. The learning sets of

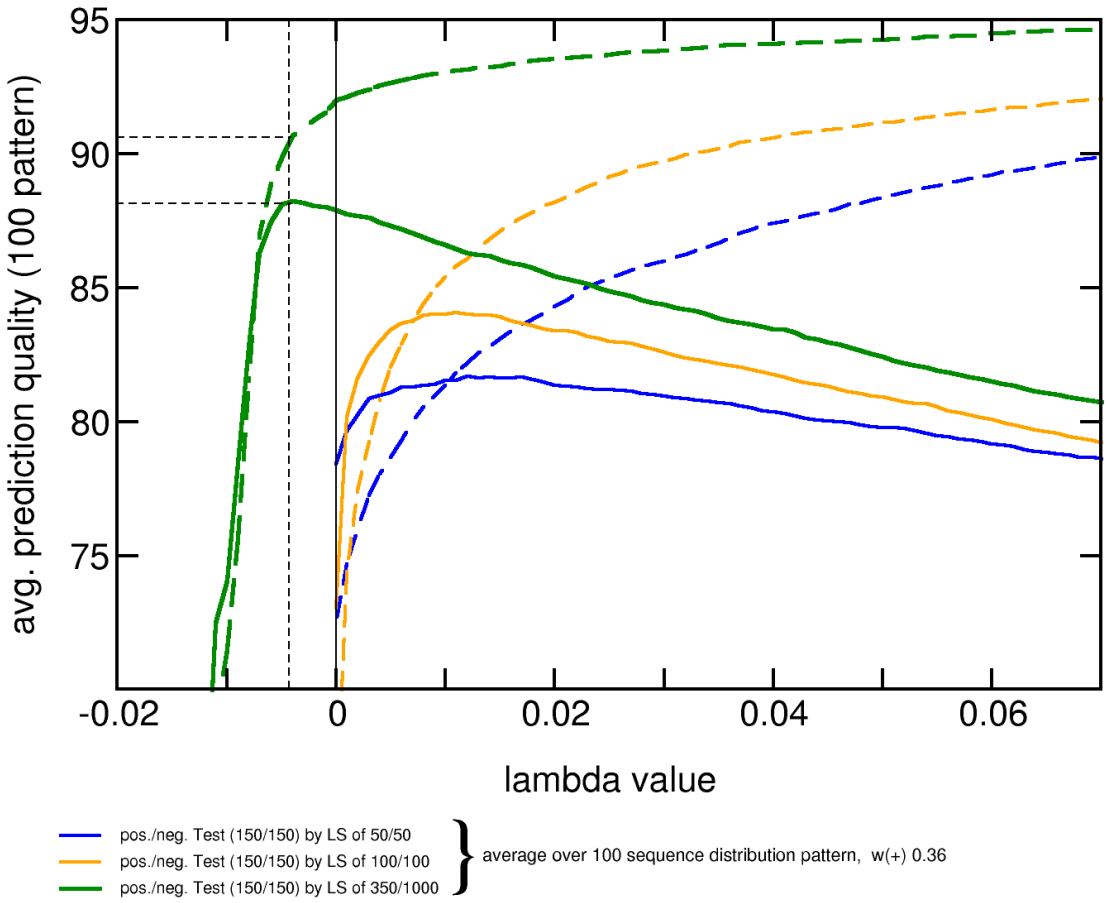


Figure 3.10: Prediction performance for the linear scoring function for different λ_w and sizes of the learning sets. Data are averages of 100 different partitions between learning and prediction set peptides. Dashed lines represent results for non-binding prediction sets, solid lines represent results for binding prediction sets.

different size should mimic different situations of true prediction. The small set is representing the situation, where few data are available to train the system. The large set is reflecting a situation, where a large number of data are available for training. To remove statistical errors, which go along with limited set size and possible unfavorable peptide distributions, average scores over 100 different peptide distributions for the learning and prediction sets are calculated. For each distribution the LSM and scoring function is computed. The results approve that for smaller learning sets the risk of learning by heart is higher, so that a larger value of λ_w can suppress this behavior. This is improving the prediction results. Due to the chosen value of w^+, the curves for binding prediction and non-binding prediction diverge. With increased values of λ_w the non-binding prediction quality is improving further, while at some value of λ_w the slope of the curve for binding prediction becomes negative. Interestingly a negative value of lambda has a positive effect for the case, where the large learning set was used. A negative lambda is increasing the influence of the larger, more pronounced parameters, which are more successful in the learning and prediction procedure.
Usually the parameter λ_w is used to suppress numerical instabilities, when the linear equation system is about to get singular. Practically it helps to improve prediction results by supressing smaller, less meaningful parameters to a higher extent. If the number of parameters is large compared to the number of learning data, a large λ_w value is improving prediction quality. If the value of this parameter becomes too large, prediction results worsen.

3.2.4 Behavior of the scoring functions in recognition and prediction

In the following three different types of classification for different scoring functions are performed.

1. A recognition of 538 binding and 538 non-binding peptides is derived for LSM, SVM and QSM.

2. For LSM, a leave-one-out crossvalidation (also called jackknife) is predicting a single peptide out of 538 binding and 538 non-binding peptides, while all remaining peptides are used for learning. This procedure was applied for all 538 binding and non-binding peptides.

3. Finally, a prediction of 538 binding and 538 non-binding peptides is performed for the LSM. A small learning set of 50 binding and 50 non-binding peptides was used. The 50 binding peptides used for learning are part of the 538 binding peptides of the prediction set, which means that for those 50 peptides a pseudo prediction is performed.

The set of 538 non-binding peptides are selected by chance from 10,000 available non-binders. The binding set contains the entire set of 538 available binding peptides. To perform the analysis the peptides are sorted by their target values, assigned by the scoring function. Different to figure 3.9, this time the binding and non-binding sets are regarded separately, yielding two branches $f^+(\vec{x})$ and $f^-(\vec{x})$ of the scoring function. Missclassified peptides of the respective set can be identified easy, where the curves cross the threshold line of zero. Binding peptides with a scoring function below zero and non-binding peptides with a scoring function above zero are missclassified.
For this analysis the SVM (green curve) and QSM (orange line) methods are applied for the 538/538-recognition, where all peptides are used for learning and recognition. The LSM method is applied for all cases described above (all blue curves). The weighting for LSM is set to $w^+ = 0.36$, while for QSM it is set to $w^+ = 0.50$. For LSM and QSM a lambda value of $\lambda_w = 10^{-6}$ was used to avoid singularities.

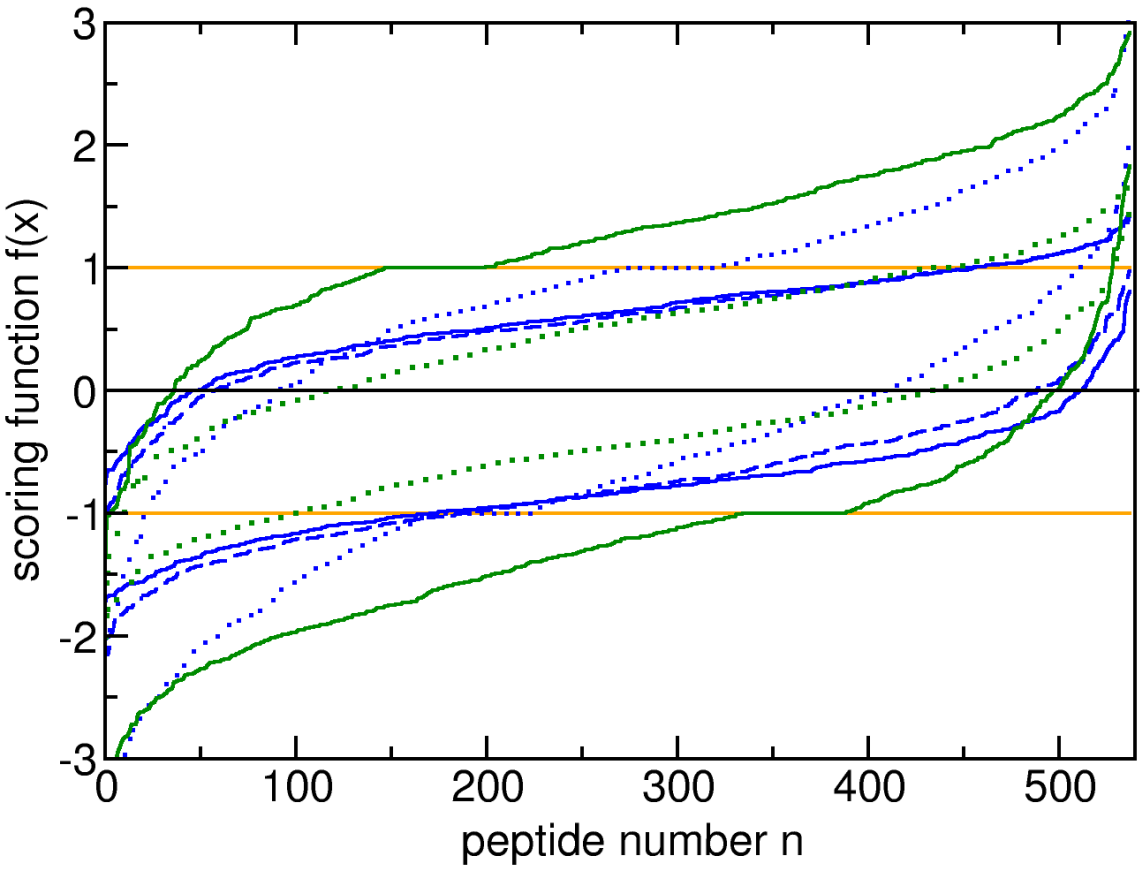


Figure 3.11: Comparison of different scoring functions separating binding and non-binding peptides in recognition or prediction sets. Solid lines represent complete recognition of 538 peptides in the learning sets. Upper curves belong to binding sets, lower curves to non-binding sets. Green curve represent SVM, orange lines represent QSM, blue lines belong to LSM. Dashed curves represents results of leave one out cross validation, dotted curves represent prediction of 538/538 petides prediction sets after training with 50/50 random chosen petides for the learning set.

As a result of the comparison between the two optimization methods, least square optimization or support vector machines are shown in figure 3.11. Both methods yield a very similar recognition quality. The SVM has a small advantage in recognizing binding peptides, while the LSM is better in recognizing non-binding peptides. For the LSM, the weighting factor w^+ can shift this relation. The quadratic scoring function (QSM) uses all linear terms plus nine amino acid categories as described in section 3.2.3. The recognition of all 538 peptides in each learning set, binding and non-binding, was performed by the QSM without any error. The scoring function of the QSM assigns for all peptides of the appropriate learning set target values, which are exactly matching the expectation values +1 for binding or -1 for non-binding peptides. This is a typical indication of learning by heart. The results for the LSM prediction jackknife or 50/50 learning show similar behavior as the recognition. It is obvious that the error rate is increased due to the fact that peptides unknown to the scoring function have to be classified. For the 50/50 set only 50 peptides of each class have been learned, but the generalization capability is still good enough to classify most of the 538 peptides of each class correctly.

3.2.5 Reassessment of the composition of the learning data sets

The support vector machine has the ability to select a subset of data in feature space to optimize the performance. Although the least square optimization does not directly offer such an option, the LSM optimization can be used to identify incorrectly recognized peptides and those peptides located in the twilight zone of vanishing scoring values. The assumed binding ability might have been assigned wrongly especially for those peptides sequences randomly generated and assumed to belong to the non-binding class. These malicious peptides may be identified by a preliminary LSM run and can be identified and eliminated before starting the actual classification approach. To investigate this, an LSM optimization with random selected 300 binding and 5,000 non-binding peptides in the learning set is initiated. The w^+ parameter is set to 0.36. All remaining 238 binding and 5,000 non-binding peptides are used for the prediction set. In the first run, 92.0% binding and 92.8% non-binding peptides from the learning set are recognized correctly. In the prediction set 90.3% of the binding and 92.5% of the non-binding peptides are classified correctly. Based on these results, 31 non-binding peptides from the learning set are removed, because their target values $f(\vec{x}^-)$ were larger than 0.7. With this modification a second run of the LSM was performed. For the learning set of the binding peptides no changes in recognition rate was observed, but the recognition rate for non-binding peptides was slightly reduced to 92.5%. In the prediction mode the fraction of binding peptides increases to 91.6%, while the rate for the non-binding peptides is now 92.2%. Hence, the overall performance for the prediction is increased compared to the first run of the LSM where all peptides of the learning set are used unfiltered.

	learning		**prediction**	
	$\mathbb{S}^+$	$\mathbb{S}^-$	$\mathbb{S}^+$	$\mathbb{S}^-$
1^{st} **LSM prerun**	92.0	92.8	90.3	92.5
2^{nd} **LSM run**	92.0	92.5	91.6	92.2

If the same sets of 300 binding and 5,000 non-binding peptides for learning and the remaining 238 binding and 5,000 non-binding peptides for prediction are provided to the support vector machine, a break down in recognition and prediction rate for binding peptides can be observed with a recognition rate of 50.6% and a prediction rate of 48.7% for binding peptides. At the same time the number for correct recognized and predicted non-binding peptides is very high with 99.5% for both recognition and prediction of the non-binding sets. This is a known problem to SVM if the used learning data sets are unbalanced, like in this case with only 300 binding but 5,000 non-binding peptides [52][17]. This problem can be overcome by simply providing several copies of the smaller data set to the learning algorithm. In this case 16 identical copies of the set of 300 binding peptides are used as new binding set to balance the data of binding and non-binding learning set. With this modification the SVM recognizes 96.0% of the binding and 92.0% of the non-binding peptides correctly. In the prediction 93.0% of the binding and 91.0% of the non-binding peptides are classified correct. In the case of the LSM the weighting factors w^+ and w^- are balancing the data sets during the learning procedure.
In conclusion it can be summarized that a well tuned least square optimization can achieve the same prediction quality as the SVM optimization.

3.2.6 Quality control via Receiver Operating Characteristics Curve

The Receiver Operating Characteristics (ROC) Curve is a sensitivity versus 1-specificity plot, which can be used as quality control for recognition and prediction results [53]. The two functions specificity and sensitivity, both depending on the threshold t, are introduced.

$$sens(t) = \frac{correct^{+}(t)}{N^{+}} \qquad \text{and} \qquad spec(t) = \frac{correct^{-}(t)}{N^{-}} \tag{3.1}$$

The variables N^{+} and N^{-} are the total number of binding or non-binding data in the learning set. The functional dependence $sens(spec)$ can be obtained by varying the threshold t used to classify a peptide given by $\vec{x}$ as binding for $f(\vec{x}) > t$ or as non-binding for $f(\vec{x}) < t$. The area under the curve for a ROC plot is an overall measure of quality.
For this examination the same 269 binding and 269 non-binding peptides were used for the learning set as described in section 3.2.1 to determine the parameters of table 3.21. For the prediction set the remaining 269 binding and 269 non-binding peptides were chosen. Just for comparison the least square optimization with quadratic terms has been added. In this case all linear terms plus two quadratic categories, hydrophilic and hydrophobic amino acids, have been chosen, while the w^{+} was set to 0.50 and a high value for the λ_w term of 10^{-2} was selected.

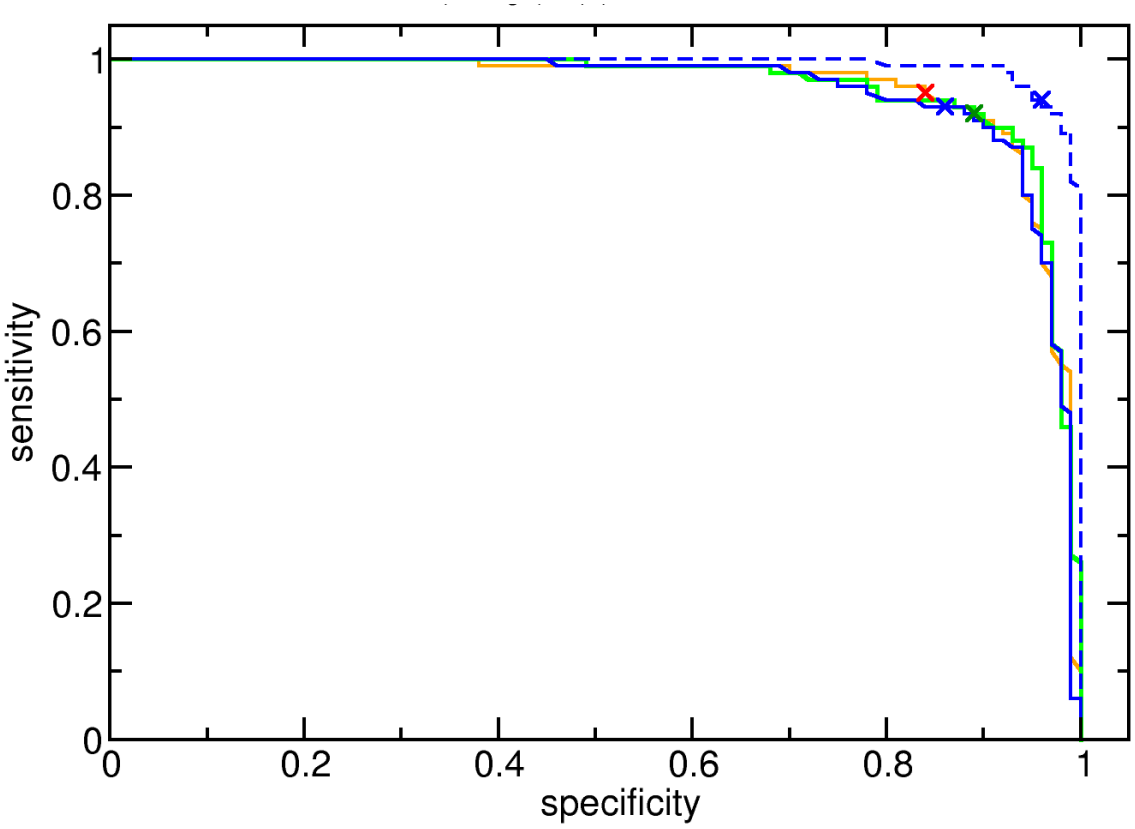


Figure 3.12: ROC plots for LSM and QSM. Solid lines are referring to prediction results, dashed lines to recognition results. The blue lines represent results for LSM optimization, the green curve is related to the SVM optimization, the orange curve is related to the QSM optimization with 20 linear plus 2 quadratic terms and a $\lambda_w = 10^{-2}$. Weighting factor w^{+} is 0.45 for LSM and 0.50 for QSM. Area under the curve (AUC) is in LSM recognition 0.9926. For LSM prediction the AUC is 0.9559, for SVM prediction the value is 0.9613 and for QSM prediction the value is 0.9592. Positions $t = 0$ are marked by crosses.

Actually the quadratic scoring function makes no sense for this application using a small learning set of 269 binding and 269 non-binding peptides.
The results demonstrate a small advantage of the SVM optimization in its ability to discriminate between binding and non-binding peptides. With the regularization and weighting parameters it is possible to tune the LSM to match the prediction quality between LSM and SVM.

3.2.7 Quality control by statistical survey

Another way to judge the quality of the LSM optimization is a statistical breakdown of recognition and prediction results by analysis of i.e. 400 different distribution patterns of the peptides divided into learning and prediction set. For this purpose the learning and prediction peptides are randomly chosen from the appropriate complete sets of peptides. Therefore 400 different sets are generated by selecting out of the total 538 binding and 10,000 non-binding peptides the desired number of peptides for learning or prediction. For this analysis two different sizes of learning sets are chosen. First 50 binding and 50 non-binding peptides are selected for the entire learning set. Second the learning set is composed out of 300 binding and 5,000 non-binding peptides. For comparison the size of the prediction sets remains in both cases the same. By chance 238 binding and 5,000 non-binding peptides are selected for the prediction sets.

		learning set 50/50		**learning set 300/5,000**	
method		**binding peptides [%]**	**non-binding peptides [%]**	**binding peptides [%]**	**non-binding peptides [%]**
LSM - linear	recognition	100.0 ± 0.0	100.0 ± 0.0	94.8 ± 0.8	91.3 ± 0.1
	prediction	78.8 ± 21.3	73.0 ± 19.5	90.0 ± 2.9	90.8 ± 0.2
QSM - quadratic [3]	recognition			95.1 ± 0.7	91.9 ± 0.1
	prediction			89.0 ± 3.9	91.3 ± 0.3

Table 3.23: Recognition and prediction statistics of 400 different runs for different size of learning sets. LSM and QSM optimization run with $w^+ = 0.45$ and $\lambda_w = 10^{-5}$. For QSM 20 linear terms plus quadratic 2 categories are used yielding 351 free parameters.

The table 3.23 displays the result for the different statistical evaluations. For comparison results of QSM with 20 linear terms plus 2 quadratic amino acid categories for hydrophobic and hydrophilic amino acids were added. Since the training with a learning set of 50 binding and 50 non-binding peptides for the linear LSM already yields learning by heart, an examination of this small learning set using QSM was skipped. It can be observed that the LSM optimization improves in prediction, if it is trained with a larger learning set of 300 binding and 5,000 non-binding peptides. The prediction and recognition rate converges, if the learning by heart phenomenon is suppressed, because the provided learning data and the number of parameters of the system reach an optimal relation. Another indicator that the obtained results for the small learning set 50/50 are suboptimal, is the high variance assigned for the prediction rates, while at the same time for the learning rates the variance is zero. Thus increasing the number

[3] using 20 linear terms plus 2 quadratic categories (hydrophob/hydrophil)

of parameters by introducing the QSM demonstrates that even for the large learning set of 300 binding and 5,000 non-binding peptides, the overall prediction rates slightly degrade, while the variance is increased slightly. It can be assumed that for the LSM with its $20 \cdot 9$ parameters an optimum is already reached.

3.2.8 Discussion and summary of section 3.2

- The self-developed least square optimized scoring function is a general approach to derive classification shown on the example of binding prediction of peptides to HLA-A0201
- The achieved prediction quality is surpassing results obtained from similar methods [54, 55, 56, 57, 58] and coming close to results of powerful, well established methods like SVM optimization or Hidden Markov Models [46, 47, 59].
- Carefully tuned training of the LSM optimization allows to achieve equivalent results in prediction compared to SVM optimization
- The LSM optimization can be manually adapted and optimized for the different prediction challenges by regularization parameters like λ_w and w^+
  The weighting factor w^+ allows evaluation of unbalanced learning data sets without a decline in prediction quality
- By using a prerun analysis, questionable peptides can be eliminated from the learning data set and therefore possibly increase the overall prediction quality. In the current analysis this makes sense since the non-binding peptides were derived by chance from protein sequences, involving the risk of potentially missclassified entries.
- For large to very large learning sets the linear LSM can be expanded by introduction of quadratic terms (QSM) - here shown on the example of amino acid categories. This allows the adaptation of the number of parameters to more complex learning patterns.
- However, the QSM quadratic terms may not always have enough flexibility. In the case of the genetic algorithm a flexible generation of quadratic features out of selected linear features is used

The presented LSM method shows a high performance for classification of HLA-A0201 binding peptides. With the two regularization parameters w^+ and λ_w the method can be tuned for different prediction challenges. It can handle classifications for small as well as large learning data sets without a significant breakdown in prediction performance. The comparison to alternative classification approaches like SVM underline that the prediction performance of the LSM is competitive to well established methods. For asymmetric learning data sets, where the number of binding peptides differ much to the number of non-binding peptides, the weighting term w^+ allows the LSM to outperform the compared SVM method. However, this handicap of the SVM can easily be fixed. For symmetric learning data sets the prediction quality of LSM comes very close to the SVM results.
One common measurement for the overall quality of prediction algorithms is the ROC curve and the related AUROC value obtained from the area under the ROC curve. The ROC allows a judgement over the classification ability when the classification threshold t is changed. Nevertheless, the discrimination between binding and non-binding peptides is mainly interesting for the threshold of $t = 0$, where the classification occurs in the normal application. Therefore,

the significance of the AUROC measure is doubtfull. Another quality measure, the Matthews Correlation Coefficient (MCC) is used in the next section, where the results for the CoEPrA 2006 competition are discussed. The MCC is correlating missclassified binders and non-binders to the number of correct classified binders and non-binders as described in section 2.3.7.

3.3 CoEPrA 2006 competition

The data provided by CoEPrA 2006 have the advantage to contain both, peptide sequences and physico-chemical derived features for all four classification tasks. The LSM scoring function can make use of both types of features. A strategy is needed to reduce the number of parameters and to prevent learning by heart if physico-chemical features shall be used. There are 643 physico-chemical features for each residue position, which yields 5787 features for a nonapeptide.

3.3.1 Ranking of the competitors for CoEPrA 2006 tasks 1-4

The following tables 3.24 - 3.27 list the ranked results of the competitors for the different classification tasks of CoEPrA 2006, which is information presented by the organizers of the CoEPrA competition. The data shown are only related to the prediction results and do not refer to recognition. The following abbreviations are used: TP,TN,FP,FN - true positive, true negative, false positive and false negative classified peptides. "Sens." and "Spec." are sensitivity and specificity as described in section 3.2.6. "Accu." is accuracy, the average rate of positive and negative correct classified peptides. "Auroc" is the area under the ROC curve and "MCC" the Matthews Correlation Coefficient.

results for classification - 1

Rank	Group	TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC	AUROC
1	Wuju Li	40	36	8	4	0.9091	0.8182	0.8636	0.7303	0.8636
2	Gavin Cawley	38	38	6	6	0.8636	0.8636	0.8636	0.7273	0.8636
3	Mehdi Jalali Heravi	36	39	5	8	0.8182	0.8864	0.8523	0.7062	0.8523
4	Matt Segall	39	35	9	5	0.8864	0.7955	0.8409	0.6847	0.8409
5	Reiji Teramoto	40	33	11	4	0.9091	0.7500	0.8295	0.6676	0.8296
6	Joao Aires-de-Sousa	38	35	9	6	0.8636	0.7955	0.8295	0.6606	0.8296
7	Fatih Amasyali	32	40	4	12	0.7273	0.9091	0.8182	0.6472	0.8182
8	Hendrik Blockeel	32	39	5	12	0.7273	0.8864	0.8068	0.6216	0.8068
9	Levon Budagyan	38	33	11	6	0.8636	0.7500	0.8068	0.6176	0.8068
10	David Farrelly	34	37	7	10	0.7727	0.8409	0.8068	0.6151	0.8068
11	Wit Jakuczun	37	34	10	7	0.8409	0.7727	0.8068	0.6151	0.8068
12	Scott Oloff	36	35	9	8	0.8182	0.7955	0.8068	0.6138	0.8068
13	Robert Kirk DeLisle	35	35	9	9	0.7955	0.7955	0.7955	0.5909	0.7954
14	Shikha Varma-O'Brien	35	33	11	9	0.7955	0.7500	0.7727	0.5460	0.7727
15	Bart De Moor	30	37	7	14	0.6818	0.8409	0.7614	0.5295	0.7614
16	Bhaskar Kulkarni	36	31	13	8	0.8182	0.7045	0.7614	0.5261	0.7614
17	Curt Breneman	37	26	18	7	0.8409	0.5909	0.7159	0.4460	0.7159
18	Elizabeth Jacob	29	33	11	15	0.6591	0.7500	0.7045	0.4108	0.7046
19	Marco Gori	36	21	23	8	0.8182	0.4773	0.6477	0.3143	0.6477
20	Effendi Widjaja	28	28	16	16	0.6364	0.6364	0.6364	0.2727	0.6364
21	Alexander Zelikovsky	31	20	24	13	0.7045	0.4545	0.5795	0.1643	0.5796
22	**Walter Knapp**	21	28	16	23	0.4773	0.6364	0.5568	0.1151	0.5568
23	Artem Cherkasov	25	24	20	19	0.5682	0.5455	0.5568	0.1137	0.5568

Table 3.24

Difficulties arise due to the small size of the learning data sets of binding and non-binding

results for classification - 2

Rank	Group	TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC
1	Levon Budagyan	33	32	6	5	0.8684	0.8421	0.8553	0.7108
2	Scott Oloff	33	32	6	5	0.8684	0.8421	0.8553	0.7108
3	Reiji Teramoto	33	32	6	5	0.8684	0.8421	0.8553	0.7108
4	Hendrik Blockeel	33	31	7	5	0.8684	0.8158	0.8421	0.6852
5	David Farrelly	36	26	12	2	0.9474	0.6842	0.8158	0.6547
6	Gavin Cawley	34	28	10	4	0.8947	0.7368	0.8158	0.6396
7	Robert Kirk DeLisle	27	34	4	11	0.7105	0.8947	0.8026	0.6158
8	Fatih Amasyali	32	27	11	6	0.8421	0.7105	0.7763	0.5575
9	B.D. Kulkarni	30	29	9	8	0.7895	0.7632	0.7763	0.5528
10	Shikha Varma-O'Brien	34	21	17	4	0.8947	0.5526	0.7237	0.4761
11	Wuju Li	29	26	12	9	0.7632	0.6842	0.7237	0.4488
12	Marco Gori	28	26	12	10	0.7368	0.6842	0.7105	0.4216
13	Matt Segall	26	26	12	12	0.6842	0.6842	0.6842	0.3684
14	Alexander Zelikovsky	38	8	30	0	1.0000	0.2105	0.6053	0.3430
15	Mehdi Jalali Heravi	22	26	12	16	0.5789	0.6842	0.6316	0.2646
16	Artem Cherkasov	16	26	12	22	0.4211	0.6842	0.5526	0.1091
17	Alexander Tropsha	25	14	24	13	0.6579	0.3684	0.5132	0.0275
18	Curt Breneman	18	18	20	20	0.4737	0.4737	0.4737	-0.0526
19	**Walter Knapp**	11	21	17	27	0.2895	0.5526	0.4211	-0.1637

Table 3.25

results for classification - 3

Rank	Group	TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC
1	Wit Jakuczun	50	40	26	17	0.7463	0.6061	0.6767	0.3560
2	Gavin Cawley	36	51	15	31	0.5373	0.7727	0.6541	0.3188
3	**E. Walter Knapp**	43	44	22	24	0.6418	0.6667	0.6541	0.3085
4	Matt Segall	40	46	20	27	0.5970	0.6970	0.6466	0.2954
5	David Farrelly	32	52	14	35	0.4776	0.7879	0.6316	0.2791
6	B.D. Kulkarni	43	42	24	24	0.6418	0.6364	0.6391	0.2782
7	Fatih Amasyali	43	42	24	24	0.6418	0.6364	0.6391	0.2782
8	Alexander Zelikovsky	41	42	24	26	0.6119	0.6364	0.6241	0.2484
9	Robert Kirk DeLisle	38	44	22	29	0.5672	0.6667	0.6165	0.2349
10	Reiji Teramoto	41	40	26	26	0.6119	0.6061	0.6090	0.2180
11	Levon Budagyan	40	39	27	27	0.5970	0.5909	0.5940	0.1879
12	Hendrik Blockeel	34	43	23	33	0.5075	0.6515	0.5789	0.1606
13	Wuju Li	43	32	34	24	0.6418	0.4848	0.5639	0.1282
14	Mehdi Jalali Heravi	32	42	24	35	0.4776	0.6364	0.5564	0.1154
15	Artem Cherkasov	38	36	30	29	0.5672	0.5455	0.5564	0.1126
16	Francisco Melo	27	41	25	40	0.4030	0.6212	0.5113	0.0248

Table 3.26

results for classification - 4

Rank	Group	TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC	AUROC
1	Gavin Cawley	13	73	19	6	0.6842	0.7935	0.7748	0.3972	0.7388
2	**Walter Knapp**	10	77	15	9	0.5263	0.8370	0.7838	0.3276	0.6816
3	Levon Budagyan	8	75	17	11	0.4211	0.8152	0.7477	0.2130	0.6181
4	Scott Oloff	5	83	9	14	0.2632	0.9022	0.7928	0.1876	0.5827
5	Hendrik Blockeel	3	87	5	16	0.1579	0.9457	0.8108	0.1508	0.5518
6	Venkat Mathura	5	80	12	14	0.2632	0.8696	0.7658	0.1388	0.5664
7	Artem Cherkasov	6	76	16	13	0.3158	0.8261	0.7387	0.1341	0.5709
8	Tom Kiehl	5	78	14	14	0.2632	0.8478	0.7477	0.1110	0.5555
9	David Farrelly	9	61	31	10	0.4737	0.6630	0.6306	0.1073	0.5684
10	Matt Segall	4	81	11	15	0.2105	0.8804	0.7658	0.1002	0.5455
11	Reiji Teramoto	5	77	15	14	0.2632	0.8370	0.7387	0.0981	0.5501
12	Wit Jakuczun	5	77	15	14	0.2632	0.8370	0.7387	0.0981	0.5501
13	Mehdi Jalali Heravi	5	73	19	14	0.2632	0.7935	0.7027	0.0518	0.5283
14	Alexander Zelikovsky	0	92	0	19	0.0000	1.0000	0.8288	0.0000	0.5000
15	Fatih Amasyali	1	85	7	18	0.0526	0.9239	0.7748	-0.0342	0.4883
16	Wuju Li	16	7	85	3	0.8421	0.0761	0.2072	-0.1076	0.4591

Table 3.27

peptides. The prediction data set cannot be used as feedback control and therefore, it was necessary to split the small learning data set into a true learning and a test prediction data set. This is limiting the number of data to train the scoring function even more.
For all classification tasks of CoEPrA 2006 physico-chemical or sequence based feature vectors can be used. For all four classification tasks the results I have submitted are based on sequence feature vectors. The w^+ weighting was optimized for each task. The LSM was either tuned by the λ_w regularization or a PCA was used to suppress selected eigenvector components and to reduce the number of parameter effectively. After the contest, in the frame of this doctoral work physico-chemical features were used for the LSM. PCA and the λ_w regularization were applied to reduce the effective number of parameters. Individual feature selection will be discussed later for the GA.
Unfortunately a program error was responsible for submitting recognition instead of prediction results for the first two classification tasks. I have fixed the error, such that the last two classification tasks were not effected.

3.3.2 Submitted individual results

3.3.2a Classification task 1

For the first classification task bare sequence based methods were used. The method is using 20 linear sequence features and no quadratic terms. For parameter control, PCA was used. Only components with the largest eigenvalues were selected. The components with the 14 largest eigenvalues were fully taken into account. The following 12 components have been faded from 100% to 0% with a linear decay. The weighting was $w^+ = 0.40$ and no λ_w was used.
Since the result published in the competition was corrupted, the true result from these settings is given here:

TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC
34	35	9	10	0.7727	0.7954	0.7841	0.5683

It is possible to improve this result with better optimized settings for the LSM. In section 3.3.3 different achievable results using sequence versus physico-chemical feature vectors together with PCA based component selection are compared.

3.3.2b Classification task 2

Also for the second task of CoEPrA a sequence vector based classification was used. Again 20 linear amino acid type sequence features and no quadratic terms have been used. Different to the approach in the first task, this time no PCA was performed. Instead the regularization parameter λ_w was used to prevent learning by heart. Further settings are: $w^+ = 0.50$ and $\lambda_w = 10^{-2}$. Again the result submitted for the competition was corrupted by the same error. The true results for the explained settings are given here:

TP	TN	FP	FN	Sens.	Spec.	Accu.	MCC
33	30	8	5	0.8684	0.7895	0.8289	0.6600

3.3.2c Classification task 3

For the 3rd classification task, sequence vectors were used as descriptors. A PCA was applied to restrict the number of parameters. From the resulting 180 parameters, using only linear terms, those 5 components exhibiting the largest eigenvalues in PCA are fully taken into account, while those components related to the following eigenvalues from rank 6 to 100 are faded with a linear decay from 100% to 0%. The parameter w^+ was set to 0.53. No lambda regularization was used. Results are shown in table 3.26 entry Knapp.

3.3.2d Classification task 4

The last classification task number 4 is difficult to learn and predict because of the high asymmetry between binding and non-binding data in the learning set. The unbalanced learning data are not the only problem. The small number of binding data makes it harder to learn and generalize the information contained in the learned. Once again sequence vectors were used for learning and the PCA was applied to analyze the eigenvalues. Only those components were selected, which correspond to the largest eigenvalues. Intermediate eigenvectors were faded. The largest 14 eigenvalues were fully selected. The following eigenvalues from rank 15 to 64 were faded from 100% to 0% with a linear decay. The weighting factor w^+ was set to 0.65 due to the small size of binding peptides in the learning set. Again only 180 components of the linear sequence vector were used and $\lambda_w = 0$. Results are shown in table 3.27.

3.3.2e Summary of submitted results

For all four CoEPrA classification tasks no physico-chemical features, but sequence based features were used, because at the time of the competition little was known of the provided physico-chemical features. In three of four cases PCA was used to identify those components, which

might have an important contribution to the learning procedure. Just in one case the classical lambda term was employed to reduce the number of free parameters. The obtained results are good enough to be ranked at least in a central position, in several cases even in the top part of the overall standings for the different classification tasks. After intense studies of the methods an improvement in prediction quality with different parameterizations can be achieved as shown in the following section.

3.3.3 Optimized predictions for CoEPrA classification tasks

Optimized parameters can improve results obtained for classification tasks 1-4. To understand how parameters are optimized, an analysis of different eigenvalue selections and lambda value regularizations has to be performed. This section is just considering the conventional established methods to improve prediction results using both sequence features and physico-chemical features. Feature selection and the results of the GA are discussed in section 3.4.
In the following the term "features" is used for physico-chemical derived features provided by CoEPrA 2006. For sequence derived features the term "sequence vector" is used. First, recognition and prediction results for classification task 1 in dependence of the number of used eigenvalues is discussed. The eigenvalues are derived by PCA and are ordered by size. Only a given number

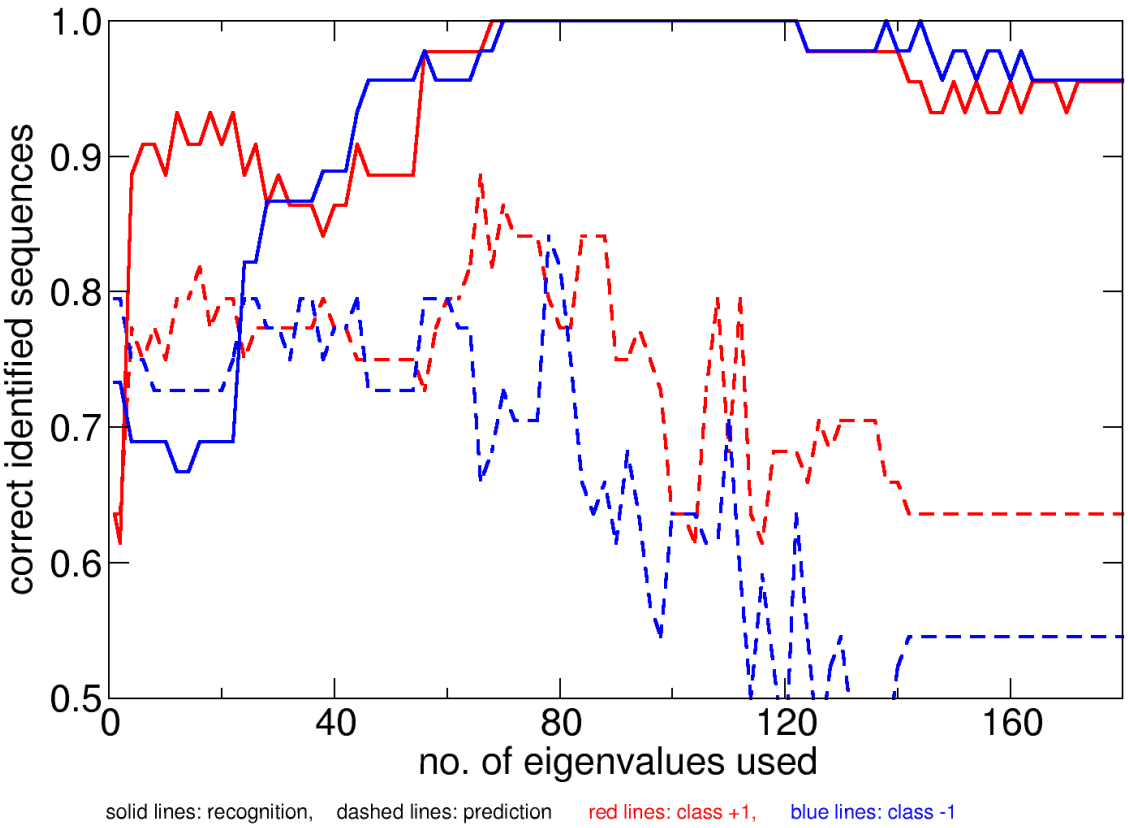


Figure 3.13: **Sequence vectors:** CoEPrA 1 classification task recognition (solid lines) and prediction (dashed lines) results for increasing number of eigenvalues used after PCA of sequence vectors yielding up to 180 parameters. A weighting of $w^+ = 0.45$, $\lambda_w = 0$ was used. Red lines refer to binding, blue lines refer to non-binding peptides.

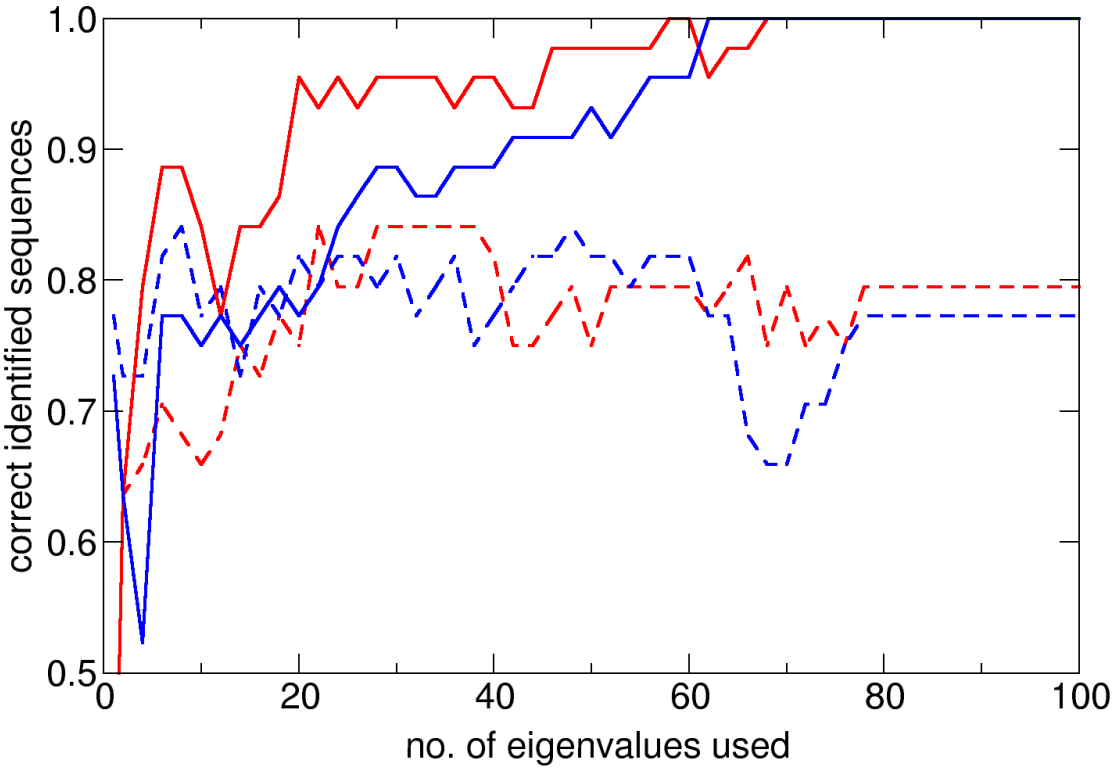
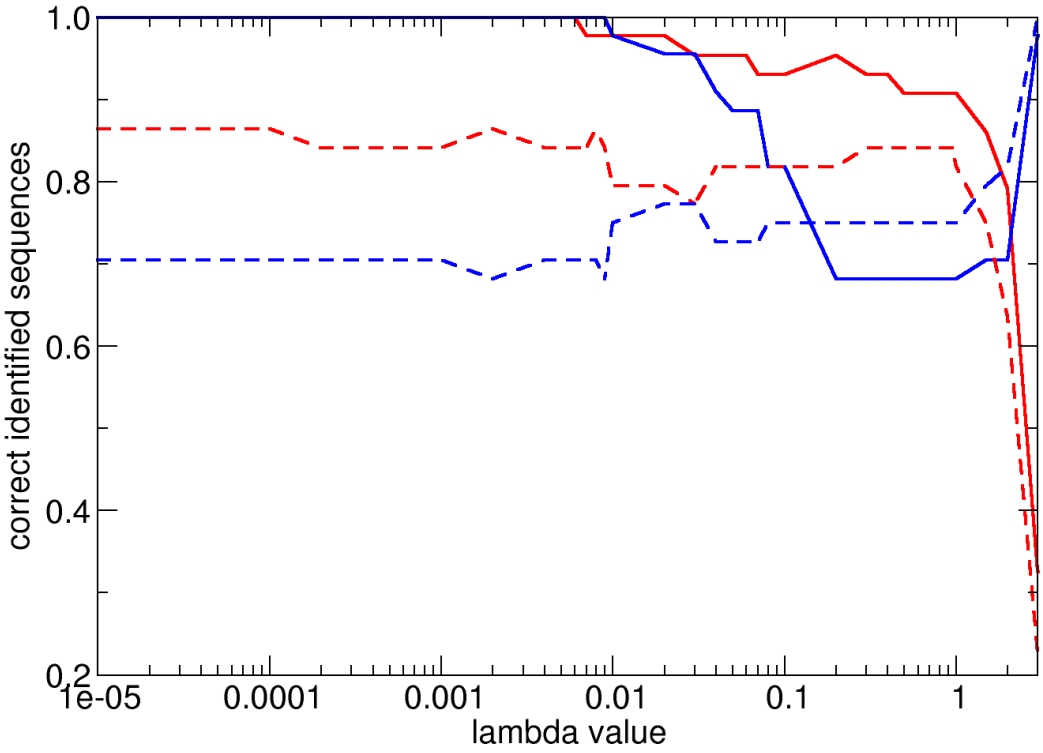
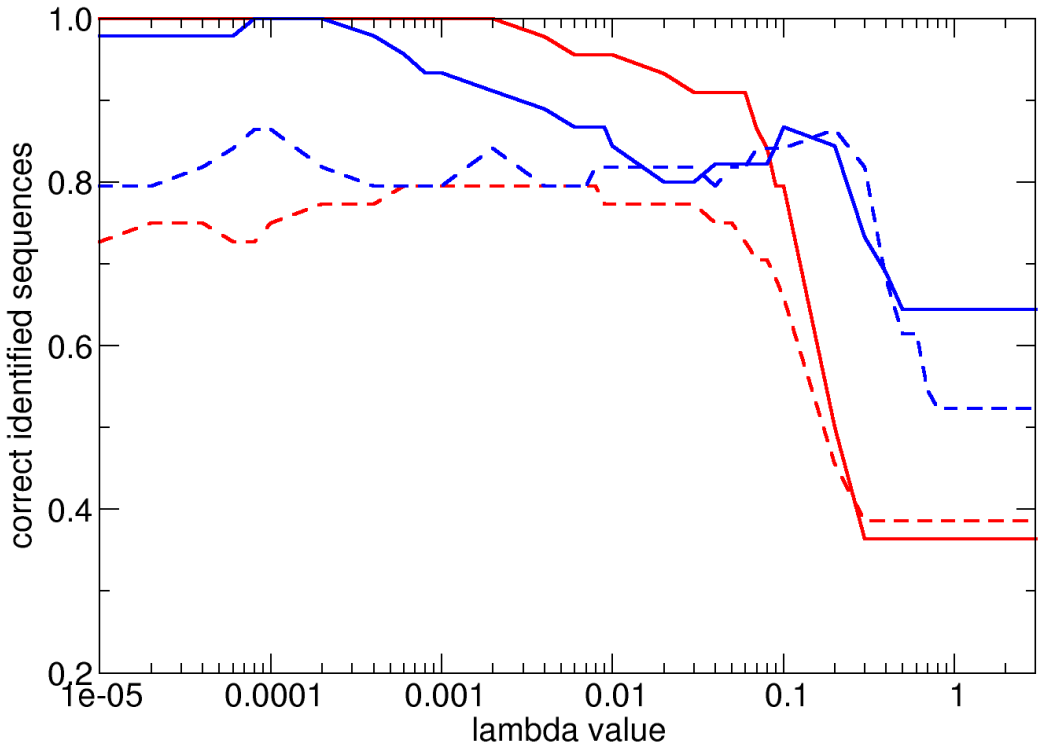


Figure 3.14: **Physico-chemical features:** CoEPrA 1 classification task recognition (solid lines) and prediction (dashed lines) results for increasing number of eigenvalues used after PCA of 5787 physico-chemical features. A weighting of $w^+ = 0.45$, $\lambda_w = 0$ was used. Red lines refer to binding, blue lines refer to non-binding peptides.

of largest eigenvalues are taken into account, removing all other values. For sequence vectors results using up to 180 eigenvalues and for physico-chemical features results using up to the first 100 eigenvalues are plotted in the two figures 3.13 and 3.14.
It is obvious that in both cases the increase in the effective number of parameters is going along with an increase in learning by heart. Nevertheless, in the case of physico-chemical features, the prediction results remain on a high level, even if a large number of eigenvalues is selected. At the same time both graphs show an increase in recognition performance of up to 100% for large numbers of eigenvalues.
Different to the submitted prediction results, here the eigenvalues are faded out by a hard cutoff. In contrast to this hard cutoff, the soft cutoff considers also components of the ordered eigenvalues from a given starting position towards a defined end position with a linear decay from 100% down to 0% of its' magnitude. Other decay functions were tested to weaken eigenvalue components, like quadratic or gaussian functions, but none of them improved the results significantly.
As mentioned before, another way to suppress unnecessary parameters is the λ_w regularization term. In figure 3.15 results for sequence vectors and for the physico-chemical features in dependence of the lambda parameter are shown. All 5787 physico-chemical features are used for the analysis in graph 3.15b.
The lambda regularization term below an value of 10^{-2} has for both cases, feature and sequence feature LSM, a stabilizing role. This means that numerically the resulting equation system is



(a) sequence vectors



(b) physico-chemical features

Figure 3.15: CoEPrA 1 classification recognition (solid lines) and prediction (dashed lines) results for different values of regularization parameter λ_w. A weighting of $w^+ = 0.45$ was used. Red lines refer to binding, blue lines refer to non-binding peptides. Using for descriptors **a)** sequence vectors **b)** 5787 physico-chemical features provided by CoEPrA

not becoming singular and at the same time learning by heart is barely suppressed. High recognition rates for binding and non-binding peptides are around 100% and indicate overfitting. If the lambda value increases above 10^{-2} recognition rates are going down and at the same time prediction rates are influenced. For sequence vector LSM the prediction rates of the two classes show a directly opposed trend with increasing lambda. Although the trend is changed for lambda values above 0.03, there is no big overall improvement to the mean prediction rate. Finally the prediction rates of binding and non-binding diverge completely above a value of 1. Interestingly for lambda value of above 1 the trends for recognition rates of binding or non-binding are coupled to the trends of the appropriate prediction rates.
In case of the CoEPrA physico-chemical feature derived LSM, effects of supressing learning by heart can be observed earlier. For a lambda value above 10^{-3} recognition rates are dropping below 100%. Prediction rates improve a little between lambda of 10^{-4} and 10^{-3}. Finally the binding prediction rate collapses followed by the recognition rate of binding peptides. Later, at a lambda value of 0.2 the rates for non-binding recognition and prediction are breaking down. Best prediction rates are achieved between lambda values of 0.001 and 0.008.

3.3.3a Magnitude of eigenvalues for task CoEPrA task 3

The dataset provided with CoEPrA task 3 is one of the most difficult to predict, since the sequence homology of the contained peptides between learning and prediction data set is high. In this example, results obtained with PCA using a strict eigenvalue cutoff are shown together with the magnitude of eigenvalues ordered by size. Once again sequence and physico-chemical

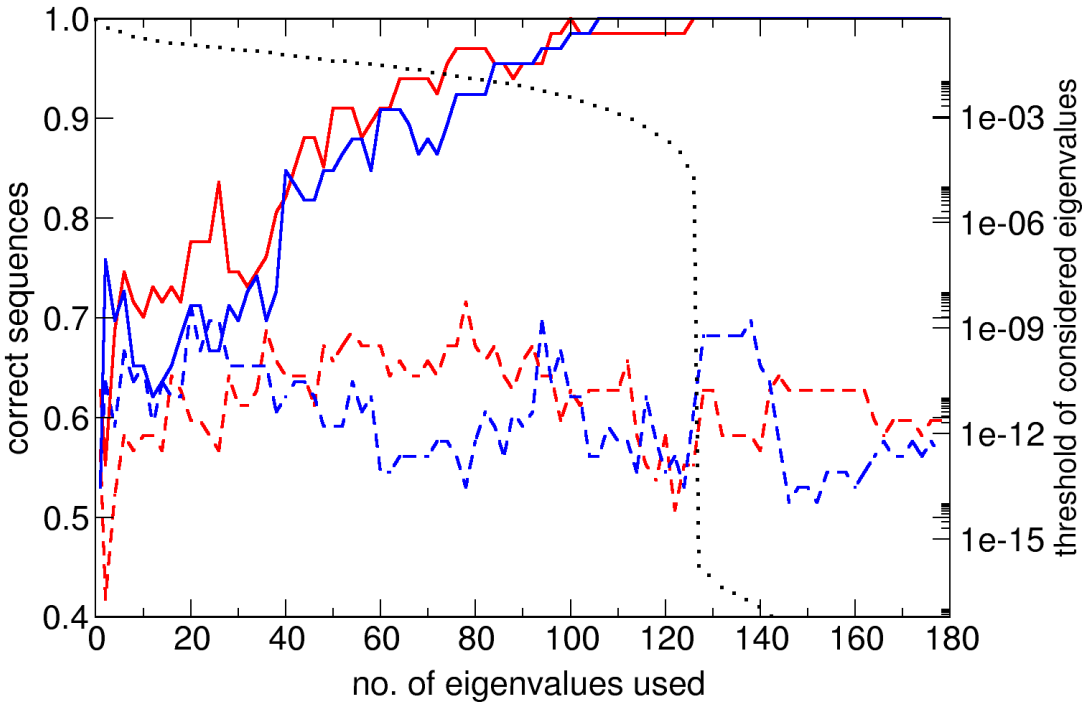


Figure 3.16: CoEPrA 3 classification by PCA and eigenvalue analysis for **sequence vectors**. A weighting of $w^+ = 0.53$ was used. Red lines refer to binding, blue lines refer to non-binding peptides. Solid lines represent recognition results, dashed lines represent prediction results. The black dotted line displays the size of the smallest considered eigenvalue referring to the right y-axis scale.

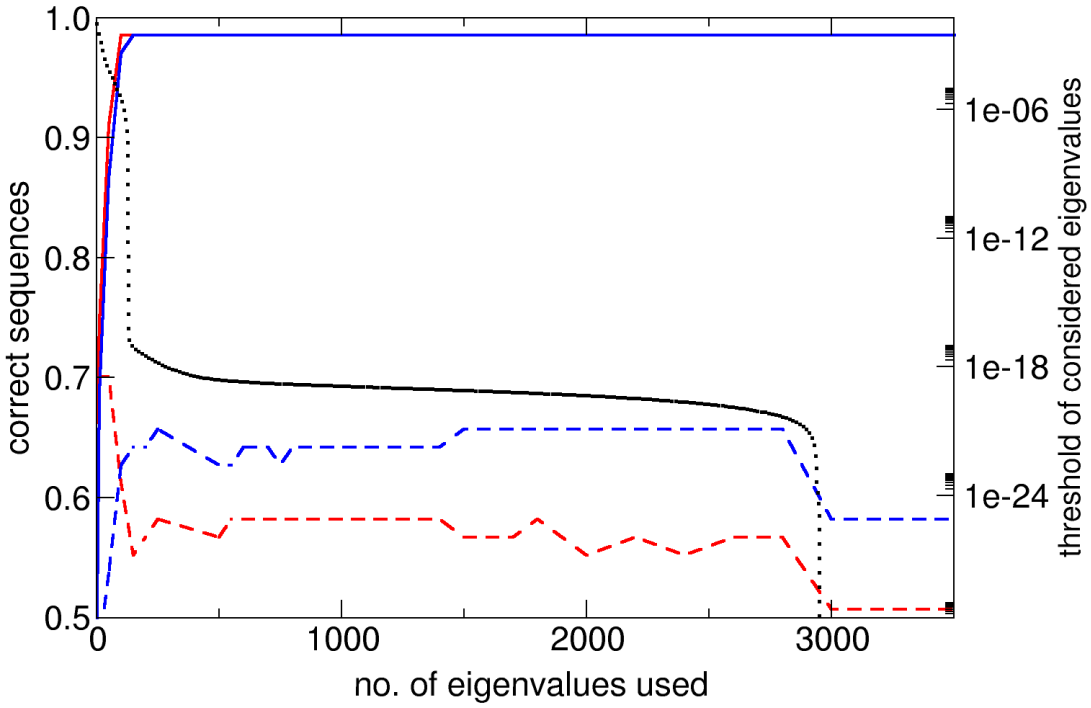


Figure 3.17: CoEPrA 3 classification by PCA and eigenvalue analysis for **physico-chemical features** showing size of assigned eigenvalues. A weighting of $w^+ = 0.55$ was used. Red lines refer to binding, blue lines refer to non-binding peptides. Solid lines represent recognition results, dashed lines represent prediction results. The black dotted line display the size of the smallest considered eigenvalue referring to the right y-axis scale.

feature vectors are selected for the LSM, using weighting factors of $w^+ = 0.53$ for the case of the sequence vectors and $w^+ = 0.55$ for the physico-chemical feature vectors.
The black curves in figures 3.16 and 3.17 display the magnitude of the eigenvalue at the current position in the stack of eigenvalues ordered by size. This corresponds to the smallest eigenvalue used to obtain the recall and prediction results. The solid colored lines are representing the recall results. The strong decay of the size of eigenvalues in both graphs 3.16 and 3.17 goes along with a recall approaching a recognition rate of almost 100%, which indicates learning by heart. This drastic change in the size of the eigenvalues indicates the threshold, where to apply the cutoff for the set of eigenvalues of the classification approach. For the sequence vector approach of CoEPrA-3 this choice seems to have little influence on the prediction quality. From the very beginning prediction rates fluctuate to a certain degree but the mean rate of binding and non-binding prediction is almost constant. Just opposite is the behavior of the physico-chemical feature vectors, where the black curve shows two slumps. Already at the first slump the eigenvalue cutoff clearly marks the highest average prediction rate of binding and non-binding sets. In this case the change in order of magnitude for the eigenvalues is correlated also to the overall prediction quality one can expect. This is also true for the second slump in the black curve corresponding to nearly 3,000 parameters. The overall prediction performance for CoEPrA-3 is with 68% - 65% lower than the one for CoEPrA-1.

3.3.3b Hand optimized results for CoEPrA classification 1-4

Table 3.28 shows results, which are obtained for different descriptors and optimization methods for all CoEPrA classification tasks. The parameters of the respective methods are manually optimized and may contain a certain bias by a feedback to the prediction results. Therefore, these results may not reflect a realistic prediction scenario in terms of an authentic prediction. Nevertheless, they are suitable to demonstrate the possibilities of the individual methods.

CoEPrA task	type	sequence cholesky	sequence PCA	phys. chem. cholesky	phys. chem. PCA
classification 1	L+	100.0	100.0	97.7	95.5
	L-	100.0	100.0	84.4	82.2
	P+	84.1	84.1	88.6	84.1
	P-	75.0	77.3	81.8	81.8
classification 2	L+	100.0	100.0	100.0	100.0
	L-	100.0	100.0	100.0	94.9
	P+	86.8	86.8	92.1	84.2
	P-	84.2	84.2	76.3	73.7
classification 3	L+	95.5	100.0	67.2	97.0
	L-	95.5	100.0	74.2	100.0
	P+	65.7	64.2	67.2	62.7
	P-	60.6	65.2	62.1	63.6
classification 4	L+	100.0	78.9	84.2	100.0
	L-	83.7	75.0	69.6	63.2
	P+	63.2	52.6	63.2	68.4
	P-	72.8	72.8	75.0	67.4

Table 3.28: Optimal tuned prediction results for all four CoEPrA tasks with different methods. All values given are percentages of correct classification. "L" denotes recall, "P" prediction, + and - denotes the classes of binding or non-binding peptides. Sequence vector descriptors are marked by "sequence" otherwise "phys. chem." marks physico-chemical features from CoEPrA. "PCA" relates to the method based on PCA and eigenvalue selection, while "cholesky" relates to the default LSM based method with lambda regularization.

3.3.3c Feature selection for the task of CoEPrA-1

Inspired by the results obtained for CoEPrA-1 classification of the first ranked Chinese group of Wuju Li, feature selection came to mind as a powerful way to optimize classification performance. In the CoEPrA classification problem 1, the Wuju group selected just 7 features from the 5787 provided CoEPrA feature set to apply for recognition and prediction. The current set of 7 features they selected was derived out of three preselected 7-features sets, which were evaluated by leave-one-out cross validation [60]. Those 7 selected features can be used for further studies on this data set. Using these features by the LSM with a weight of $w^+ = 0.40$ for the CoEPrA-1 data a prediction rate of 90.9% for binding and 84.1% for non-binding peptides can be achieved, yielding an MCC value of 0.75. The group of Wuju Li was achieving an MCC of 0.73 with these features in CoEPrA task 1.

An optimal feature set should contain condensed information to discriminate binding from non-binding classes but skip any redundant or contradicting information of the learning data. These 7 features, which had been selected by Wuju et al. are representing the CoEPrA-1 data on a very optimal level. Still those selected features are general enough to predict the unknown data from the prediction data set with high performance. The figure 3.18 is demonstrating this fact by adding random features to the set of the seven preselected features.

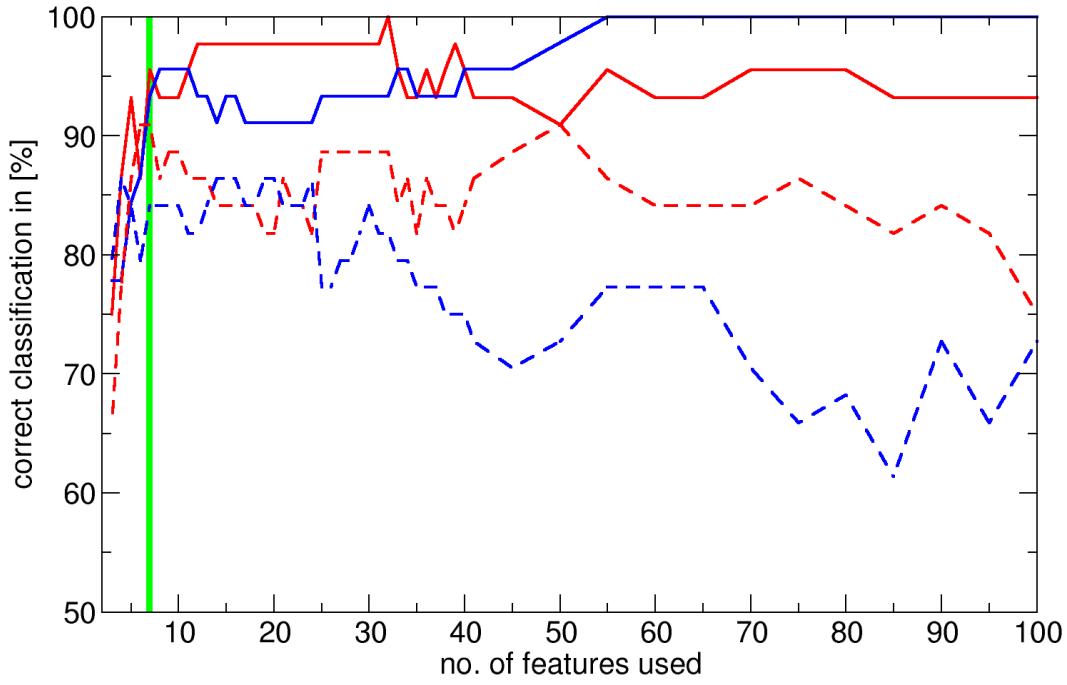


Figure 3.18: Feature selection made on base of the 7 features of Wuju et al. for CoEPrA-1. The exact Wuju feature set is marked by the green bar. For less features in the set single features have been removed from the Wuju original set. For more than 7 features random features have been added to the original set. Solid lines mark recall, dashed lines mark prediction. Red lines represent binding and blue lines represent non-binding classes. Weighting is set to $w^+ = 0.40$ and lambda is set to $\lambda_w = 10^{-9}$.

Adding random features to the selected feature set of Wuju et al. is step by step decreasing prediction performance. There are fluctuations and in some cases adding a single feature is temporarily improving the prediction performance. After adding a number of features to the set, recognition quality is approaching values close to 100% and is indicating overfitting. Although the weighting factor w^+ has been hand optimized for this problem, it is easy to see that between binding and non-binding classes is a gap in recognition and prediction quality. This is increasing with the number of added features, since for a total feature number smaller than 10 the difference between binding to non-binding peptides is vanishing.

3.3.4 Discussion and summary of section 3.3

- The LSM is suitable to deal with sequence derived and physico-chemical derived features. Since the latter one allows peptide independent description of any kind of drug molecules, physico-chemical derived features are more general.
- To handle the features used in the descriptors and to avoid overfitting, two approaches have been studied in this section: the lambda regularization term to reduce the number of effective features, and the PCA based eigenvalue analysis with the selection of most significant eigenvalue components to solve the linear equation system.
  The eigenvalue analysis seems to be more analytical, while the tuning of the lambda parameter is mainly empirical. Anyhow, the eigenvalue analysis is not always the method of choice. The way how to fade out eigenvalues was empirically explored. The linear decay fade out turned out to work best for the given CoEPrA examples.
- Knowing the prediction results it is obviously easy to optimize parameters of different approaches yielding good results. The demanding task is to optimize the results without touching or analyzing the results of the prediction set.
- Each CoEPrA classification task has different demands. While task 1 offers the easiest to predict data set, the sets 2 and 3 show higher homology in peptide sequences between binding and non-binding set and finally the set no. 4 is highly asymmetric between binding and non-binding peptides in the learning set. All CoEPrA tasks possess small training data sets.
- The remarkable results of the participating group of Wuju et al. for the classification task 1 was demonstrating the power of feature selection. This idea is the basis for the study done in the following sections.

In this section the power of the LSM method was demonstrated for the classification tasks of the competition CoEPrA 2006. The obtained results can be compared with results from other participants. Results of the LSM based approach are ranked in the upper third to the midfield of participants. The different classification tasks of CoEPrA address different situations of prediction scenarios: The first classification task is an easy to predict data set of peptides, where the trained patterns of the learning data set are easily applied to the prediction data set. Most peptides of this set can easily be separated by classes. In the second CoEPrA task sequence homology between the classes of binding and non-binding peptides is high. Therefore the peptides are more difficult to classify and the learning data set is small. The third classification task is also difficult to predict and the range of results obtained from different competitors in this competition suggest that the best achievable prediction performance is much below the performance for classification task 1. The fourth task has the highest demands for an classification algorithm. The learning data set is very asymmetric, where the number of binding peptides is very low and therefore hard to learn. This is reflected in the low performance achieved by all competitors for this task.
The LSM method can achieve average to good results for all four classification tasks, if sequence based features are used. The usage of the 5,787 physico-chemical features makes the reduction of the number of parameters crucial. PCA eigenvector selection or lambda regularization were used to control the number of effective parameters. With appropriate tuning of the LSM using weighting factor w^+ and regularization parameter λ_w or an effective cutoff of selected eigenvalue components as well as the right choice of sequence vector or physico-chemical features, good prediction results can be achieved for all four CoEPrA tasks. Nevertheless, no single approach

alone based on the LSM generates good results for all four tasks. Feature selection by an heuristic algorithm, like GA, is expected to further optimize the prediction performance for all four classification tasks.

3.4 Feature selection on the example of the CoEPrA tasks

As illustrated in the last section, feature selection is a powerful tool for classification tasks. One competitor demonstrated for the CoEPrA classification task 1 that a handful of features can be used to achieve top results. Therefore, task 1 seems to be a good starting point for the application of feature selection. Before applying feature selection the performance of each single feature in the provided set should be examined. Fortunately, a feature combination of seven features provided by the competitors Wuju et al. is known to work very well. Thus these features can be compared to the rest of the 5787 features from the CoEPrA set.

3.4.1 Single feature performance

As given by the formulas in equation 2.35a and 2.35b the performance for single features can be derived. To use this kind of formalism implies the use of the feature normalization as described in section 2.3.8. This will provide information about each features' total recognition performance as well as the number of correct recognized binding and non-binding molecules from the learning set. Figure 3.19 shows the performance of all given 5787 single features for CoEPrA-1,

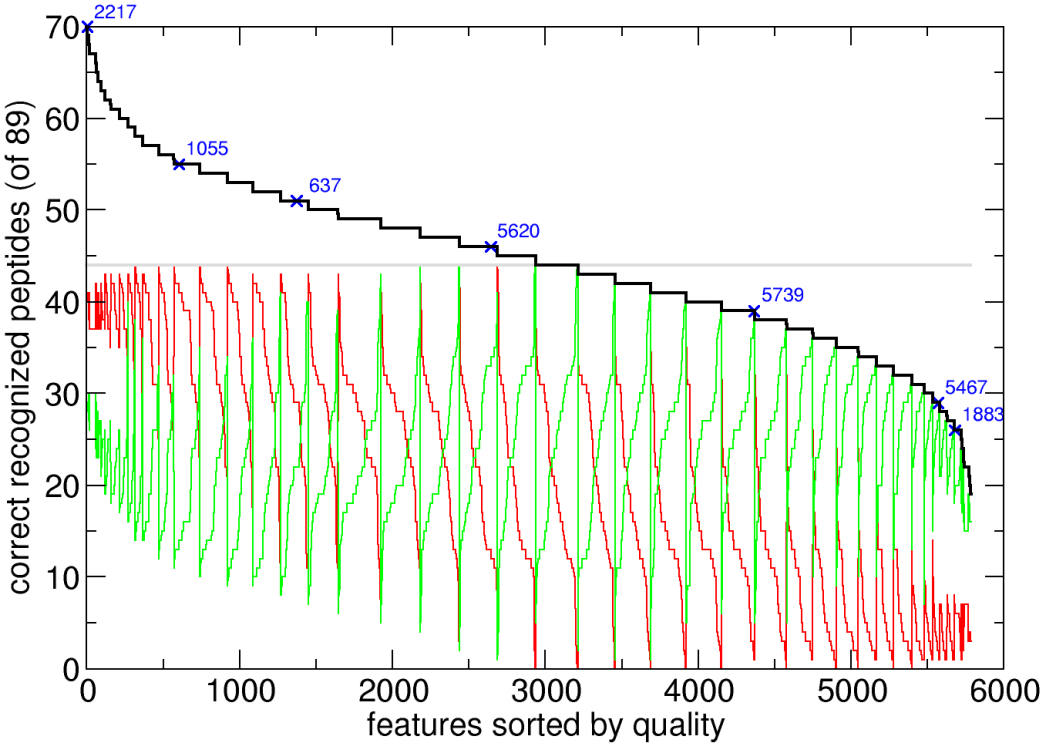


Figure 3.19: Single feature recognition performance for CoEPrA-1. All 5787 features are sorted by their individual recognition performance. Green and red curves mark the number of correct classified peptides of the non-binding or binding class, while the black curve gives the overall recognition performance for each feature. The maximum achievable performance is 89 correct recognized binding plus non-binding peptides (44+/45-). The highest actually observed number of correct recognized peptides for a single feature is 70 (41+/29-). Features whose overall performance is at or below the gray line can be inverted to gain a minimum recognition quality of 50%. Features marked by blue crosses are members of the Wuju feature set of seven features.

highlighting those 7 features of the Wuju feature set.
One single feature in CoEPrA-1 (the one on the very left side of the graph 3.19) can recognize 70 peptides from the learning set. The features in the graph are ordered by their recognition quality (overall recognition first, followed by recognition of binding peptides) such that features on the left hand side are above average. Features whose overall recognition performance is at or below the gray line are below an average recognition of 50% of all peptides from the learning set. Therefore, features at or below that line can be inverted, to obtain a minimum recognition rate of 50%. It can be recognized that almost all features are able to classify at least some peptides from both classes, binding and non-binding correctly. The ratio of correct classified peptides between the classes is varying. Binding peptides are slightly better recognized by features possesing a high overall recognition rate.
The features of the feature set of Wuju et al. are distributed over the complete width of the graph yielding 4 features which are above the 50% recognition rate. The three remaining features can be inverted in their performance. Just focussing on the four best performing features from the Wuju feature set is not improving the overall prediction rate, if the LSM is used for classification. The combination of all features from the Wuju feature set is contributing to the achieved prediction performance and it seems that weak performing single features are contributing in a specific way to optimize the classification task. It can be imagined that there are some peptides, which are hardly classified correctly by good performing features alone. The right choice of weaker performing features might add just the classification ability to recognize such problematic peptides. Combining features regarding their individual recognition performance alone seems not a sufficient criterion to obtain a well performing set of features. The idea would be just to combine excellent performing single features to create a feature set but the information which peptides are classified correctly by which feature using several features simultaneously is hidden and may vary. Combining features, which contradict in recognition can have contrary effects. Good performing features might recognize always the same type of peptides correctly, but fail for a number of peptides, which are recognized only by those features possessing a weak overall performance. The determination of the single performance of features alone is not enough to create good feature sets.
Another option is to see how often each peptide from the learning data set is correctly identified by the features of the given feature set of Wuju et al. How good is the set of features covering the complete learning set and which of those peptides from the learning set are not classified correctly? These questions are considered in the following graph 3.20.

Those peptides from the learning set, which are missclassified if one uses the Wuju feature set for LSM classification are marked by a red frame in the graph. There are 3 wrong classified binding and 4 wrong classified non-binding peptides. The binding peptides are on average more often correctly recognized by the 7 features from Wuju et.al. This corresponds to the observations from figure 3.19. In the worst case non-binding peptides are correctly identified by 2 different single features, while the binding peptides are correctly identified by at least 4 different single features. The optimal recognition rate of all 7 features is achieved for 2 peptides from the binding set, while the best result obtained for the non-binding peptides is 6 correct classifying single features in the case of 2 peptides.
Although it is hard to transform these observations into a fixed rule, one can conclude that it is helpful that each peptide in the set should be identified correctly by several features at the same time. Possibly it is an indicator of learning by heart if all the peptides from the learning set are recognized by many features from a given feature set.

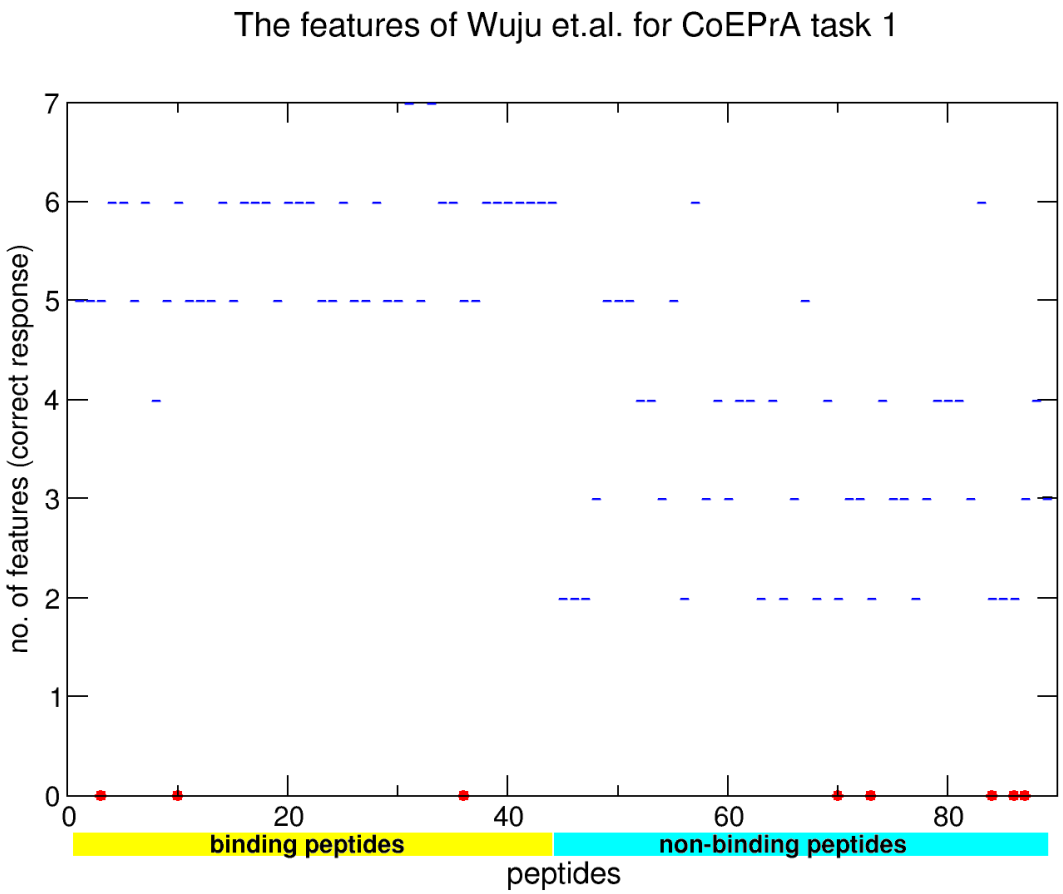


Figure 3.20: Number of single features from the feature set of Wuju et al., which correctly identify different peptides from the learning set of CoEPrA-1. Seven features are included in that feature set. The learning set contains 44 binding and 45 non-binding peptides. Binding peptides are shown on the left side, non-binding on the right side of the graph. Those peptides which are missclassified by the entire feature set using the LSM method are marked with a red dot.

3.4.2 Feature preselection by the antipode algorithm

From the data shown in the last section it can be concluded that for feature selection not only the single feature performance is an important point, but also the variety of different recognized peptides between different features plays a role. The antipode algorithm introduced in section 2.3.10 is performing a preselection of features based on recognition quality and recognition diversity. Before applying the antipode algorithm the feature set is extended by adding all quadratic features to the linear features. This will increase the size of the entire feature set enormously from round about 5,787 features to more than 16 million features.
While the individual features from the extended feature pool are normalized and evaluated regarding their recognition performance, features without a contribution (those which have a constant value for all peptides of the learning set) are discarded. Features with a recognition rate below 50% are inverted and all features below the given quality threshold parameter are removed. This cutoff threshold is one parameter for preselection and has to be above or equal 0.5 since all features with a recognition rate below 0.5 are inverted such that the rate will become $1.0 - x$. In the classification task of CoEPrA-1 a threshold value of 0.6 was used. The last

step before applying the antipode algorithm is to group the features into groups F^+, F^- and F^0 containing features of different recognition performance discriminating between recognized binding and non-binding peptides. With respect to the equations 2.36, the values for α^+ and α^- are chosen to be $\alpha^+ = 0.14$ and $\alpha^- = 0.01$ to influence the size of the three feature groups in CoEPrA-1. The features within each feature group are sorted by their recognition performance using the bucket sort algorithm. In the following the different individual features in the feature sets F^0, F^+, F^- are shown before and after applying the antipode filtering for the combined set of all linear and all quadratic features. All features must fullfill the quality criterion of a minimum recognition rate of 0.6 (= 60% of 89 peptides from the learning set). In fig. 3.21 the feature distribution before applying the antipode filtering is shown.

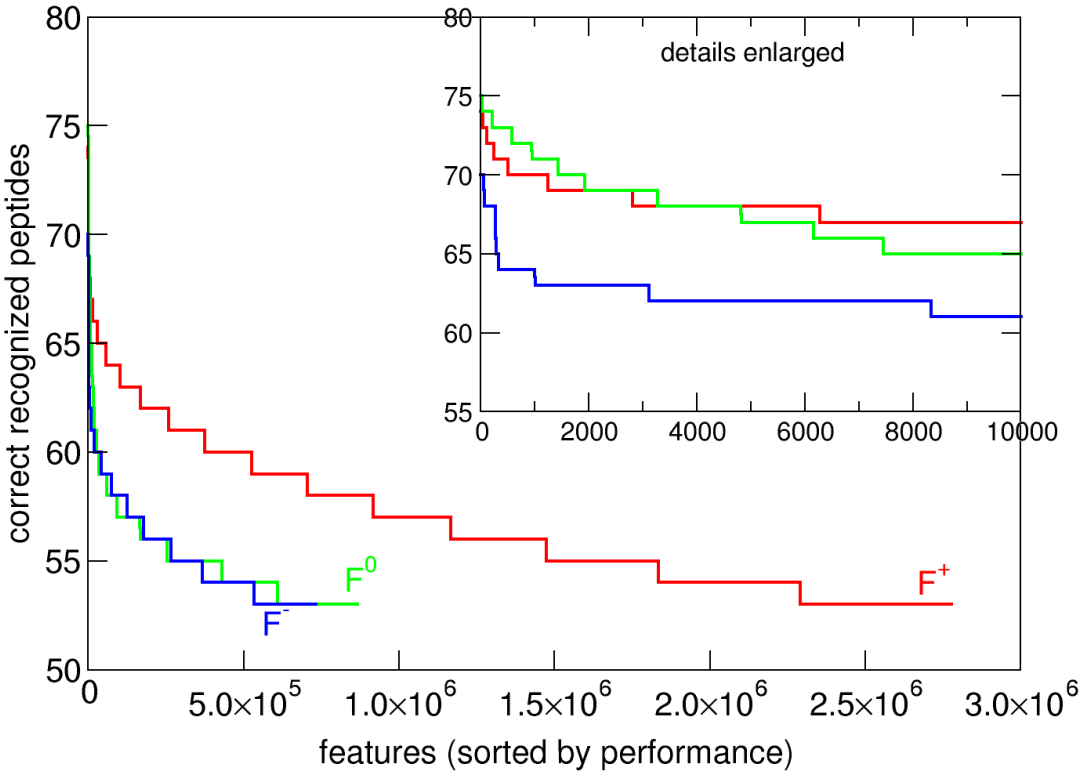


Figure 3.21: Performance of features of three feature groups F^+ (red), F^- (blue) and F^0 (green) for CoEPrA-1 for all linear and quadratic features with a minimum recognition rate of 0.6. All features are ordered by their recognition performance. Feature group boundaries are shifted by $\alpha^+ = 0.14$ and $\alpha^- = 0.01$ to achieve a balanced size of all three set F^0, F^+, F^-. Details of the features with highest performance are enlarged in the upper right corner.

The initial size of the three feature groups can be controlled by three parameters. The quality threshold is defining how many features are overall selected, the two α-parameter define how many feature from the initial F^+ and F^- group are shifted to the neutral feature group F^0. One strategy for the CoEPrA datasets was, to take care that all three feature groups do not become to different in size. Nevertheless, it can be recognized from fig. 3.21 that the three initial feature groups are different in size. Without the control by these three parameters, differences in the size of the feature groups would be magnified.
The feature group F^+ is by far the largest group, containing 2,776,971 features, while F^- contains

733,519 features and F^0 finally possesses 865,195 features. Thus the overall number of features containing all linear and quadratic features is 4,375,685. It is interesting to see that the groups F^0 contains the largest number of very good performing features, followed by group F^+. The feature group F^- contains fewer excellent performing features.
Using the antipode filtering, one can equalize size and performance of the three feature groups by choosing the parameters for the antipode algorithm for each feature group separately. The similarity thresholds τ_+, τ_- and τ_0 influence the resulting size of the appropriate feature groups after the antipode filtering. In the example of CoEPrA-1 the values for the similarity threshold are chosen to be $\tau_+ = 0.1$, $\tau_0 = 0.2$ and $\tau_- = 0.3$. All those features, which possess a too large similarity to already selected features of the new reduced sets are excluded from these new reduced sets with respect to the τ thresholds. The new reduced feature sets become the new antipode filtered feature groups F^+_{AP}, F^-_{AP} and F^0_{AP} which will be from here on simply referred to as F^+, F^- and F^0. Figure 3.22 shows the resulting feature groups after antipode filtering.

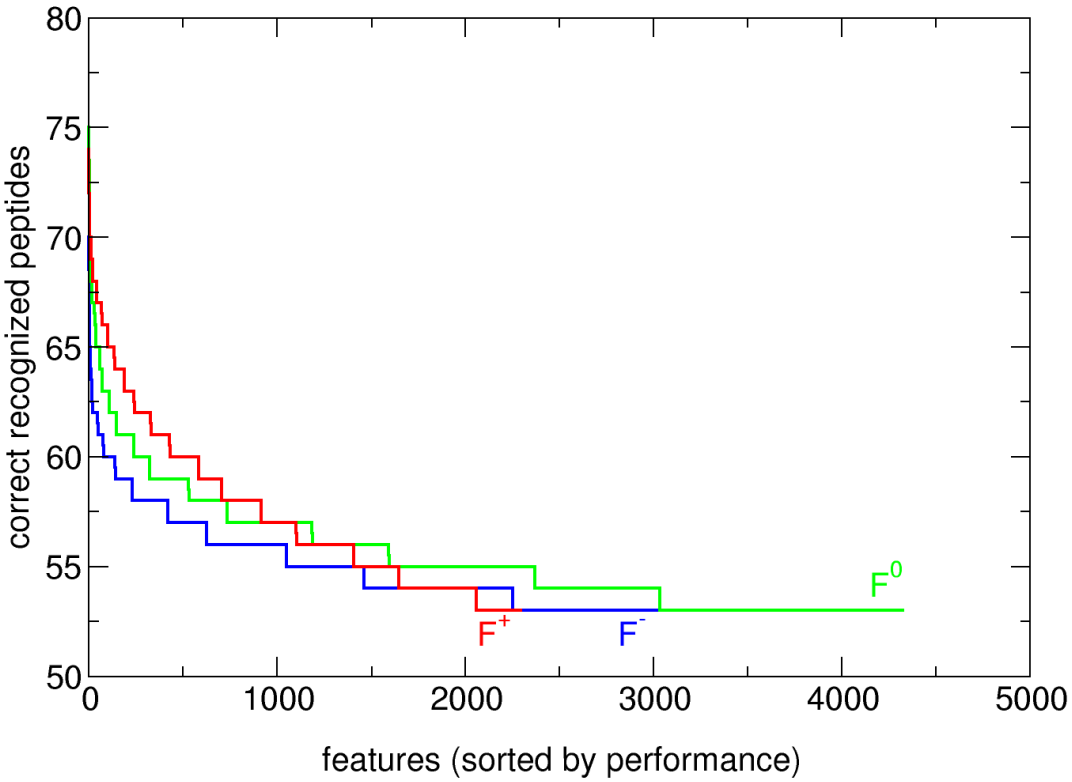


Figure 3.22: Feature performance after antipode prefiltering of features of three feature groups F^+ (red), F^- (blue) and F^0 (green) for CoEPrA-1 for all linear and quadratic features. Similarity thresholds are $\tau_+ = 0.1$, $\tau_- = 0.3$ and $\tau_0 = 0.2$. All features are ordered by their recognition performance.

It is easy to see that the total number of features per feature group is drastically reduced. All feature groups are closer together with respect to the recognition performance of their features. The group of binders preferring features F^+ is now on the top, followed by the feature groups F^0 and F^-. There are now 2295 features contained in F^+, 3057 features in F^- and 4321 features in F^0.
It is interesting to study the composition of elements from the different feature groups. It can be

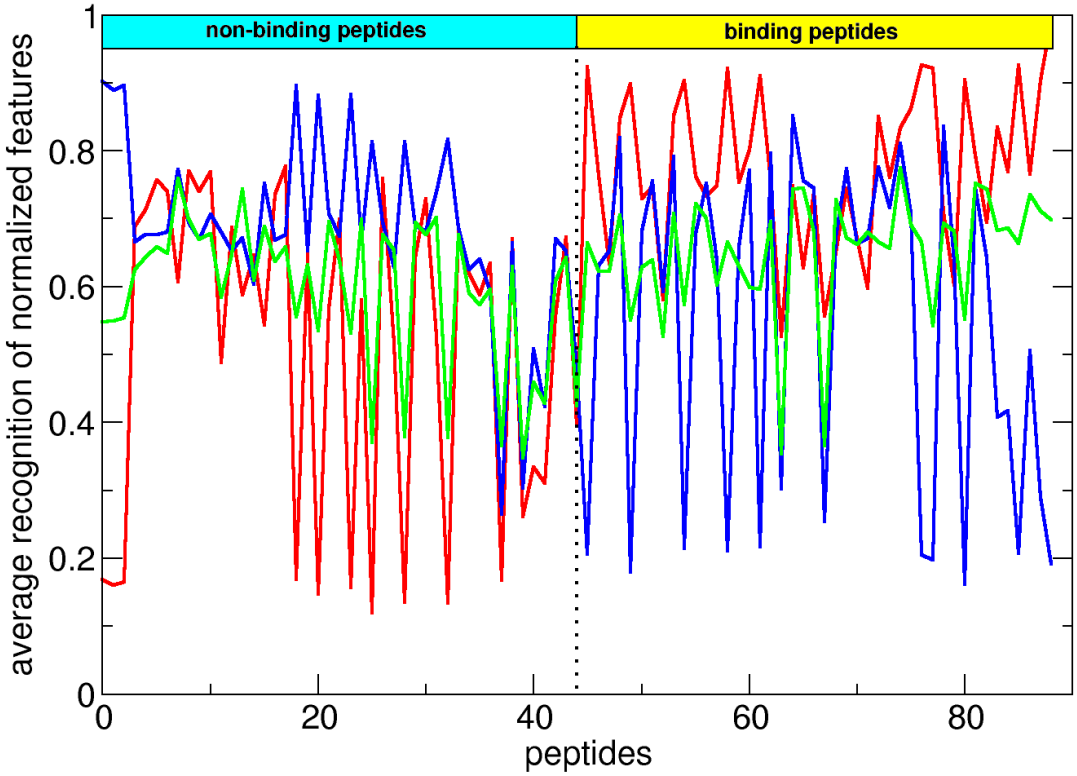


Figure 3.23: Features of different antipode prefiltered feature groups recognize peptides of binding and non-binding classes differently (CoEPrA-1). Average recognition by different features is calculated for each feature group by dividing the total number of correct hits by the number of features in the appropriate feature group. Feature groups F^+ (red), F^- (blue) and F^0 (green) show different recognition strength for binding or non-binding peptides as to be expected.

expected that features of F^+ recognize peptides from the binding set better than features from the other two groups. The non-binding peptides should be recognized best by features from the group F^-. The feature group F^0 finally should be universally good recognizing both classes of peptides. To get a comparable plot, the number of features of a given feature group identifying a molecule correctly has to be normalized by dividing them by the total number of features per group. This is shown in graph 3.23. The relation of the three curves to each other are basically as to be expected from the postulated behavior of the feature groups specific features. The majority of binding peptides are best recognized by features from the F^+ group, while the majority of non-binding features are best recognized by features from the F^- group. The features from the F^0 group are recognizing binding and non-binding peptides equally well.

3.4.3 Feature selection by the Genetic algorithm (GA)

Table 3.29 gives an overview of reasonable parameters for the GA and antipode algorithm and the results obtained with these settings for all 4 CoEPrA classification tasks. In the following MCC refers to the MCC value of the prediction set and LMCC refers to the MCC of the learning set. As displayed in table 3.29 one major problem is the large variation within the results of the 200 individuals in the final generation reflected by the difference in MCC of prediction between best and worst individual. The best results presented here are good enough to achieve a top position

	CoEPrA classification tasks			
	1	**2**	**3**	**4**
parameters of antipode feature preselection				
group threshold α^+	0.14	0.30	0.10	0.05
group threshold α^-	0.01	0.50	0.10	0.50
similarity thres. F^+	0.10	0.10	0.20	0.20
similarity thres. F^-	0.30	0.20	0.20	0.50
similarity thres. F^0	0.20	0.15	0.20	0.60
feature quality threshold	0.60	0.58	0.60	0.60
parameters of GA				
no. of GA cycles	200	200	200	200
no. of F^+ features	2	2	2	1
no. of F^- features	2	2	2	1
no. of F^0 features	3	2	3	1
Generation size in individuals	200	200	200	200
seed of random numbers	24	4	7	7
mutation rate	0.40	0.40	0.40	0.40
reproduction rate	0.20	0.20	0.20	0.20
remodulation rate	0.20	0.20	0.20	0.20
crossing over rate	0.20	0.20	0.20	0.20
no. of dropped best individuals/gen.	1	1	1	1
parameters scoring function				
lambda λ_w	1E-06	1E-06	1E-06	1E-06
weighting w^+	0.50	0.50	0.50	0.45
learning set splitting	4	4	4	4
prediction results				
best individual (MCC)	0.750	0.709	0.353	0.307
best individual (LMCC)	0.710	0.744	0.444	0.327
worst individual (MCC)	0.435	0.053	0.023	-0.171
worst individual (LMCC)	0.697	0.556	0.479	0.317
best ranked Competitors				
1^{st} ranked (MCC)	0.730	0.711	0.356	0.397
2^{nd} ranked (MCC)	0.727	0.711	0.319	0.328

Table 3.29: Parameters which have been optimized for CoEPrA tasks 1-4 yielding the shown results for best and worst performing individual of the final 200. generation from a GA run. The random seed for the GA has been chosen such that a typical performance out of 30 runs was obtained. The last two rows list the best results obtained from all participants during the CoEPrA competition 2006.

for the different tasks of the competition, but the worst performing individual(s) of the same generation would lead to the last rank. In all runs to optimize the parameters yielding competitive results, it was not possible to obtain in the final generation mostly top performing individual feature sets. Furthermore it turned out to be a real challenge to select those individuals, which are performing very well, or to exclude those which are, very weak performing individuals, due to learning by heart.

3.4.3a Evaluation of the GA results

In case of the first CoEPrA classification example the correlation between different measures shall be compared. Therefore, the 200 individuals of the final 200th generation are examined. The achieved average MCC value for the test prediction sets <MCC(aT)> is compared to the MCC value of the complete learning set MCC(tL) or to the MCC value of the true prediction set MCC(tP), respectively. Average value means that the test prediction set for all four fractions of the entire learning set was used. Furthermore, the peptide distribution for the four fractions was altered 20 times resulting in 20x4 different test learning and test prediction sets (see section 2.3.11). During this procedure the information of the true prediction was not used at any time. Only at the stage of the postanalysis the true prediction data set was used to analyze the results.

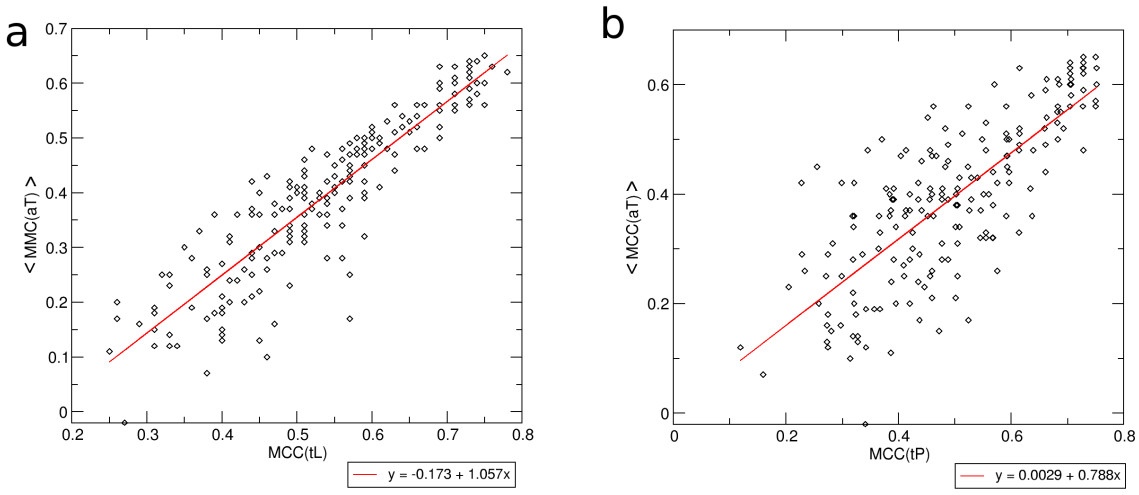


Figure 3.24: Correlations of the MCC value for the complete learning set/ true prediction set versus the average MCC value for the test prediction sets of CoEPrA-1. **a)** MCC of complete learning *MCC(tL)* correlated to MCC for the average test prediction *<MCC(aT)>* **b)** MCC of true prediction *MCC(tP)* correlated to MCC average of the test prediction *<MCC(aT)>*

During runtime of the GA a score Q is evaluated in every cycle by adding a weighted minimum value of the MCC for the test predictions to the average MCC value of the test predictions as shown in eqn. 2.42 in section 2.3.11e. To justify this reweighting the correlation between the minimum MCC value of the test predictions compared to the value of the MCC for the complete learning set or true prediction set have to be considered. The graph b in figure 3.25 yields a typical examples of the correlation of CoEPrA-1 for the final 200th generation after completing the GA. Although the result in that graph refer to the final generation of individuals, this correlation behavior already exists for individuals during the optimization procedure of the GA cycles. The minimum MCC value of the test prediction is a reasonable candidate to be used for filtering out good from bad individuals during the post processing. For this classification a possible cutoff threshold of the minimum MCC values is indicated in the graphs a,b by the blue lines.

In figure 3.26a,b the performance increase during the evolution of generations is shown. The evolution of the performance of the different generations is converging for recall and prediction

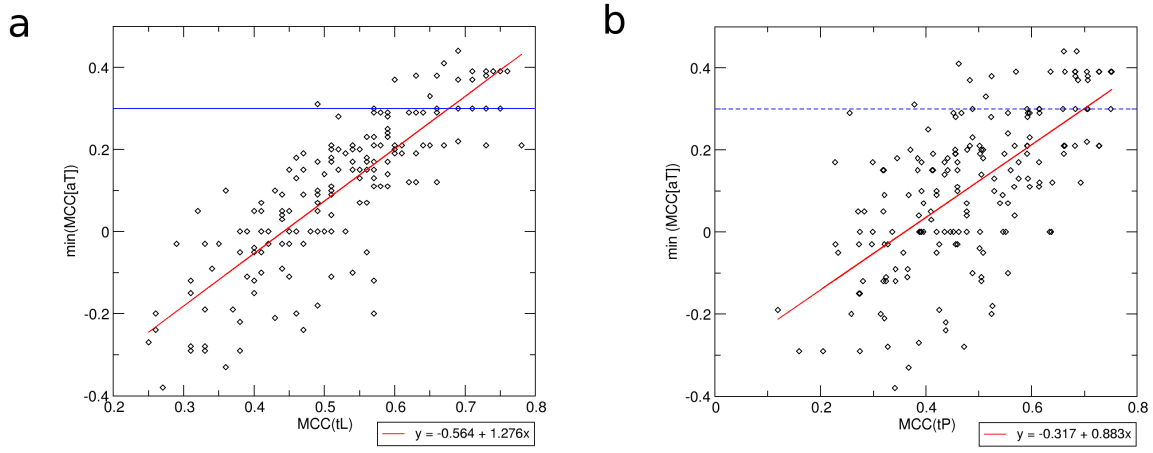


Figure 3.25: Correlations of the MCC value for the complete learning set/ true prediction set to the minimum MCC value for the test prediction sets of CoEPrA-1. The blue lines indicate a possible cutoff threshold to exclude bad performing individuals from the final selection of CoEPrA-1 during post processing. **a)** MCC of the complete learning *MCC(tL)* correlated to minimum MCC for the test prediction *min(MCC(aT))* **b)** MCC of true prediction *MCC(tP)* correlated to minimum MCC of the test prediction *min(MCC(aT))*

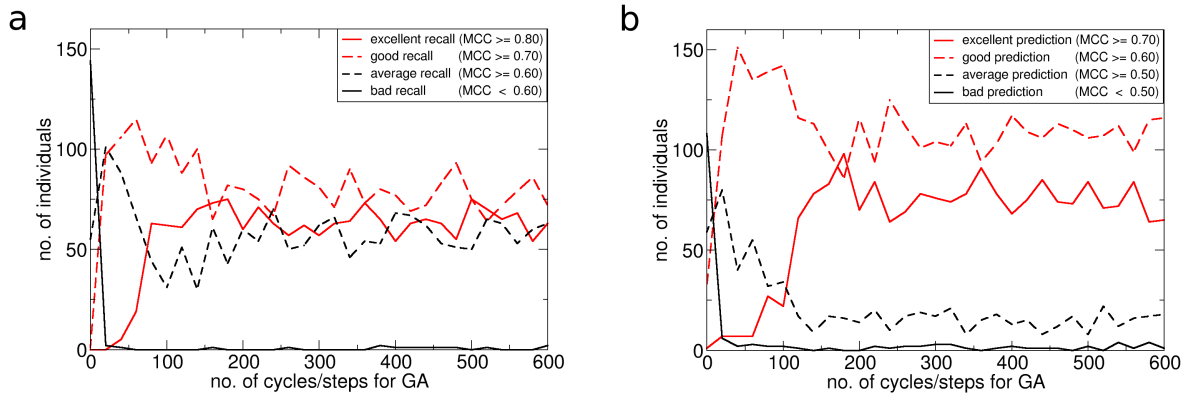


Figure 3.26: Time evolution of recognition and prediction performance in the GA on the example of CoEPrA task 1. Enrichment of good performing individuals is occurring after 100-200 cycles of the algorithm for both cases. Fluctuations in the content of generations after convergence is caused by the settings for the genetic operations. Measures for quality categories in recall and prediction are slightly different since a certain high MCC value for an individual in recall is easier to achieve compared to the same MCC value in prediction. **a.** time evolution for recall (learning) **b.** time evolution for prediction.

within 100 to 200 generations. There are fluctuations in the composition of the generations, due to the random nature of the process to obtain a new generation that vary these results only to a negligible extent. The main conclusion from this is the observable enrichment of successful individuals achieved by the GA.

3.4.4 Discussion and summary of section 3.4

- Before feature selection is used a quadratic feature expansion is performed to generate the products of all linear features. Meaningless features are removed and each feature is evaluated regarding its' recognition ability. Features with recognition rates below 50% are inverted. Grouping the features into feature groups F^+, F^- and F^0 is based on recognition rates of the learning classes + and -. Features are sorted by their performance and features below a quality threshold given for each feature group are discarded.
  The antipode algorithm removes all features from a feature group, which are too similar to the growing group of dissimilar features. The starting point are good performing single features and the comparison proceeds to features with decreasing recognition performance.

- The assembly of the 3 feature groups can be controlled by specific parameters. It is possible to counterbalance the 3 feature groups to each other to prevent strong asymmetries in size. Feature reduction as done by the antipode algorithm is crucial for performance and quality reasons before entering the genetic algorithm (GA)

- Feature selection from these three feature groups is performed by the GA. At the end of the GA a number of feature sets called individuals will be returned. Correlation between test prediction performance and complete learning performance of the entire learning set is strong. There is still a good agreement between the test prediction performance and the performance of the true prediction. The test prediction set is part of the complete learning set and a stronger correlation to the learning set has to be expected.

- The GA evolution plots show an enrichment of good and excellent performing individuals over the number of genetic cycles. This is one key demand for the GA.

- CoEPrA classification tasks show a diverse behavior regarding the GA. In most cases the GA produce individuals, which are performing well compared to the results of the competitors. Nevertheless, each generation contains a number of individuals, which are intensively learning by heart. They are showing good performance only with respect to the learning set. Without knowledge of the prediction set it is hardly possible to filter out learning by heart individuals reliably. In some cases a generation does not contain good performing individuals.

- Feature selection is working well for CoEPrA classification task 1. All other tasks are causing more or less problems. Final generations can be very heterogeneous regarding the true prediction performance of the individuals.

The GA is supposed to provide a generation of individuals, which is enriched by good performing individuals. This can be observed for the different CoEPrA task 1-4. The features available to the GA are preselected by the antipode algorithm to improve the convergence of the GA. It can be observed in the GA evolution plots that the GA is converging within 100 to 200 iteration cycles. The resulting best and worst individuals for the four tasks of CoEPrA are shown in this section. Parameters of table 3.29 have been individually optimized. For CoEPrA task 1 and 3 the first or second rank respectively could have been reached in the competition, if the best performing individual had been selected. For the CoEPrA tasks 2 and 4 the 4th or 3rd rank respectively could have been reached by choosing the best performing individual. Nevertheless, the weakness of this process is the variance in performance for the individuals from each generation and the need to find out, which individual performs well. The lower end of performance achieved for CoEPrA tasks 1-4 by the results of the GA is given as worst performing

individuals in table 3.29. A post-analysis of the individuals of a generation is crucial to separate good from bad individuals. Especially the CoEPrA task 2-4 require a post-processing, since more weak performing individuals are generated per generation compared to task 1. Different indicator values like the minimum MCC value of the test prediction show a good correlation to the MCC of the true prediction and can be used for the selection of good individuals.

3.5 Post-processing of GA optimized individuals of the final generation

Figure 3.26 in the previous section demonstrates that each generation, even the final generation after 200 applied genetic cycles, is contaminated by a number of individuals, which possess a poor performance in prediction. Since there is no unique correlation between the performance in recall and prediction for the same individual, it is hard to detect those individuals, which should be discarded. In fact learning by heart can emphasize individuals, which are excellent performing in recall but have a weak performance in prediction. The main reason to apply a post-processing or post-analysis after the GA is to filter out those individuals, which are good performing in prediction even without any knowledge of the true prediction results. In other words an automatism has to be found to select good behaving individuals in all or at least most test cases such that in a real prediction scenario individuals with good performance are found.

3.5.1 Selection of successful individuals

For each individual a number of quantities based on the MCC values and statistical derived quantities from the MCC values can be used as indicator values as described in section 2.3.12. Most important is that the MCC of the true prediction set cannot be used for the decision process but only for the control of the final success of the applied selection. The idea behind this is that those individuals, which are learning by heart to some extend can be detected after analysis of the given indicator quantities. In the following table 3.30 an excerpt of a typical final generation from CoEPrA task 1 after 200 GA cycles is given. The individuals of the last generation are ranked by the average MCC value of the test prediction set. Only highest ranked individuals are shown. Individuals with a certain behavior regarding the indicator values are highlighted in the table.

Indicator values are average MCC values of test learning sets (aL) and test prediction sets (aT). Those sets are altered 20x4 times due to 20 different peptide distributions to the learning sets of four blocks. For each distribution in 4 blocks one block can become test prediction set, while the remaining 3 blocks are used for test learning. The mean values are calculated out of all 80 MCC values for test learning and test prediction sets. Furthermore, a minimum MCC value of the test predictions $min(< aT >)$ and the MCC variance $var(< aT >)$ of the test prediction can be calculated. The MCC of the complete learning set (tL) and true prediction set (tP) are given on the right hand side of the table. The tP-value cannot be used for the procedure to select successful individuals.

Those individuals which are suspected to do learning by heart should be excluded from the list. In the table entries are highlighted in red, which possess values possibly indicating learning by heart. This is the case if the average test prediction value (aT) is above the average value of test learning (aL). Another indicator is the MCC for the complete learning set (tL). If the performance of an individual to recognize the complete learning set is much higher than the performance of other individuals from the same generation this could be a sign of learning by heart. Furthermore should the MCC performance for the complete learning set be lower as the average MCC of the test learning sets from the same individual. Not shown in this table is the influence of the minimum value $min(< aT >)$, which always should be high with respect to other individuals. Also the distance of the minimum value $min(< aT >)$ to the test prediction average aT can be helpful to characterize the individuals reliability. It can happen that individuals are eliminated from the list (as in row 2 of table 3.30 for individual #46), which actually are performing well in prediction but fullfill the criteria of a suspect to learning by heart. This

	MCC statistics over 20x4 subsets			complete sets			
	$\frac{3}{4}$ sets	$\frac{1}{4}$ sets				missclassified	
indiv.	**aL**	**aT**	**min<aT>**	**tL**	**tP**	**learn**	**pred.**
2	0.84	0.82	0.64	0.845	0.683	7	14
46	0.81	0.82	0.64	0.798	0.705	9	13
1	0.81	0.81	0.64	0.798	0.705	9	13
5	0.81	0.81	0.64	0.798	0.705	9	13
37	0.81	0.81	0.64	0.798	0.728	9	12
0	0.84	0.80	0.55	0.845	0.683	7	14
8	0.81	0.80	0.48	0.798	0.728	9	12

Table 3.30: Excerpt of best ranked individuals of the final generation for the example of CoEPrA task 1 using feature sets composed of 2 F^+, 2 F^- and 3 F^0 features with $\lambda_w = 10^{-10}$. The ranking order is given by the average MCC of the test prediction sets (aT). Individuals with conspicuous indicator values are highlighted in red. Those values could indicate learning by heart behavior. This is the case if the average MCC of the test learning sets (aL) is smaller compared to the average MCC of test prediction sets (aT). Furthermore are comparable high MCC values of the complete learning set (tL) suspicious. The column (tP) yields the MCC value for the true prediction.

is acceptable as long as bad individuals are eliminated for sure.
This procedure works fine for the classification task 1 of CoEPrA 2006, but often fails for classification tasks as for instance CoEPrA 3 or 4. Task 3 and 4 are in general more difficult to predict. As demonstrated by the results of all competitors in the contest, the achievable MCC values for the prediction sets of those tasks are very low (between 0.3 and 0.4 compared to MCC values above 0.7 for task 1 and 2). This is also reflected by the results obtained from the learning data set. Learning, test learning and test prediction performances are very low for these tasks, which requires completely different criteria to handle the detection of outliers.

3.5.2 Molecular data set similarities to guide the selection of individuals

In a more recent approach the compositions of learning and prediction data sets are compared with respect to the selected features of each individual. How similar are the different sets of data used to train the algorithm compared to the sets of data for prediction through the eyes of the features included in the current feature selection? Here it is possible to differentiate between each class (binding = + and non-binding = -) and the type (learning or prediction) such that four subsets of data can be compared. In the following, learning sets are marked by the letter "L" and prediction sets are marked by an "P", while a following plus or minus characterizes the class to which the set belongs to. Data similarity is derived by the following procedure as described in detail in section 2.3.13.
The table 3.31 refers the results for learning and prediction data sets using the complete number of physico-chemical features or the entire number of sequence derived features. Large values indicate high similarity of the compared data. All values have been normalized according to equation 2.43.

The first lines listed in the table refer to the (normalized) self-similarity of the learning sets and the prediction sets, respectively. One trend which can be observed from the given similarity

		physico-chemical features				sequence based features			
		task 1	**task 2**	**task 3**	**task 4**	**task 1**	**task 2**	**task 3**	**task 4**
learning set									
+	+	0.804	0.778	0.745	0.736	0.302	0.161	0.181	0.168
+	-	0.736	0.779	0.725	0.706	0.168	0.125	0.132	0.117
-	-	0.713	0.815	0.726	0.707	0.137	0.141	0.133	0.133
prediction set									
+	+	0.788	0.753	0.742	0.739	0.249	0.131	0.170	0.157
+	-	0.695	0.714	0.717	0.708	0.124	0.102	0.122	0.110
-	-	0.668	0.706	0.709	0.705	0.111	0.125	0.123	0.125
learning - prediction									
+	+	0.793	0.758	0.740	0.722	0.261	0.123	0.162	0.122
+	-	0.699	0.725	0.717	0.704	0.134	0.117	0.124	0.113
-	+	0.731	0.766	0.726	0.711	0.152	0.111	0.131	0.119
-	-	0.682	0.749	0.713	0.702	0.102	0.108	0.113	0.117

Table 3.31: Average similarity of feature vectors from data sets of the four CoEPrA tasks differentiating between physico-chemical and peptide sequence derived feature vectors. Similarity values are mean values of all set to set pairings. A value of +1 means identity. Random matches correspond to a value close to zero. All data containing information of the prediction set cannot be used for an in-depth analysis, since class association is not available in true prediction scenarios. This information is just to get an impression how the prediction sets compare to the learning sets.

values, is that for almost every case the self-similarity of binding peptides from learning and prediction sets are higher than the comparable values of the corresponding non-binding sets. This is the case for both types of features used, physico-chemical and sequence based features. This is easy to understand if one argues that the variation can be higher for unspecific petides as it is for binding petides. Only in the case of CoEPrA task 2 the values between binding and non-binding data set show the opposite behavior - and this is valid only for the sets using physico-chemical derived features. Another striking characteristics of the present values is the significant higher similarity of physico-chemical derived feature sets compared to the sequence based feature sets. physico-chemical features reflect the similarity of different amino acids with comparable characteristic better than sequence coded features, which depict a change in the sequence with always the same distance in feature space between any pair of amino acids Hence, similarity measures in terms of sequence encoded features are not directly comparable to physico-chemical derived features.

For the self-similarity values of the prediction set the same trend for binding to non-binding sets can be observed. This is consistent with the previous conclusion. If results of the different CoEPrA tasks are compared to each other, task 1 shows the highest self-similarity for the binding set of physico-chemical features and sequence based features. Even for the prediction set the self-similarity of the binding peptides is the highest in CoEPrA task 1. This classification task 1 is indeed the one, which works best for most classifiers. There is only little deviation between the data of the other CoEPrA tasks. Even for the prediction set self-similarity comparison is not providing new understanding.

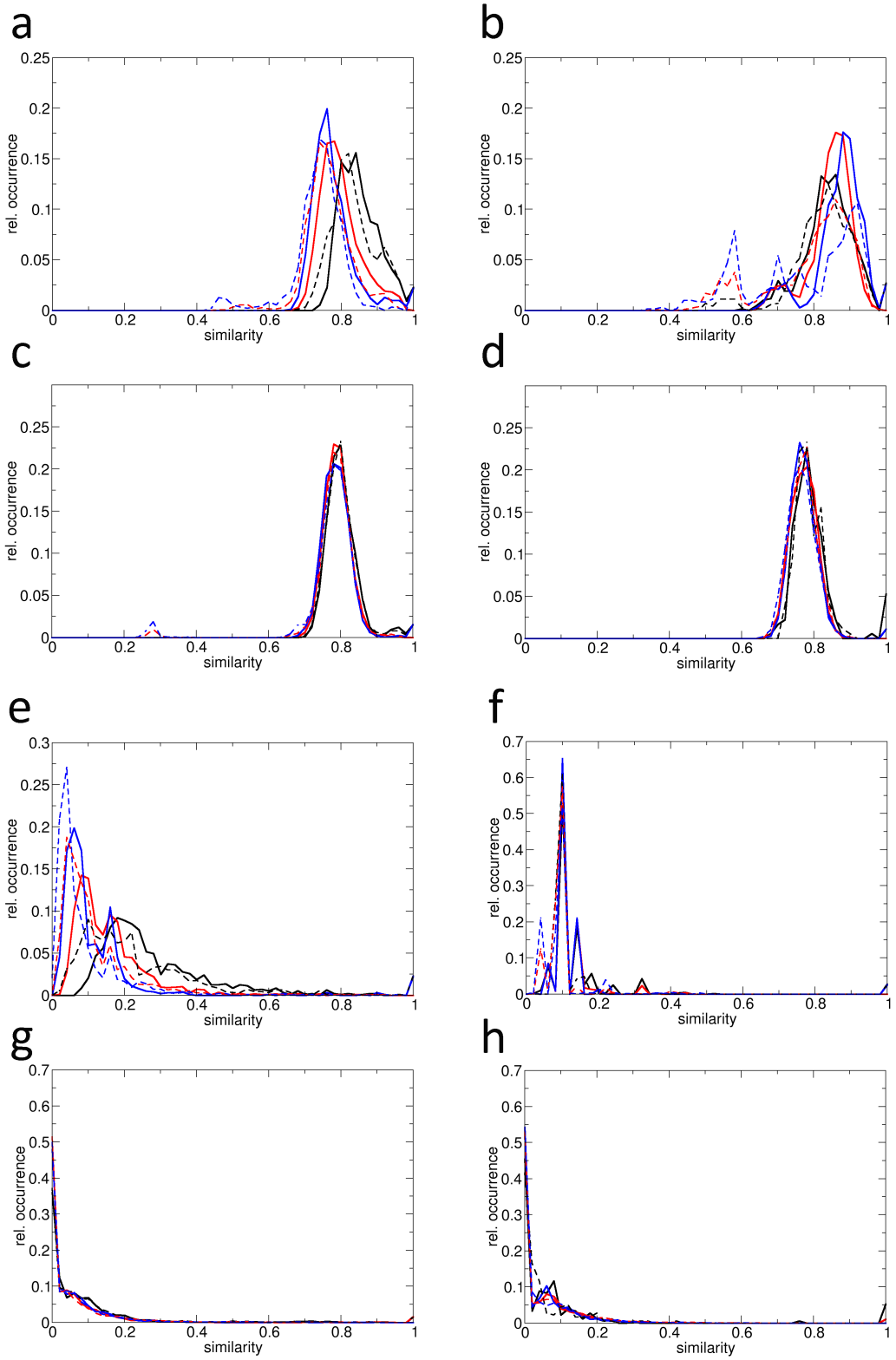


Figure 3.27: Histograms of the similarity distribution between pairs of peptide sets using different feature sets: **a-d:** for physico-chemical -, **e-h:** for sequence based - features each on the CoEPrA tasks 1-4. Color code used: L+L+ black solid, L-L- red solid, L+L- blue solid, P+P+ black dashed, P-P- red dashed, P+P- blue dashed.

The similarity between learning and prediction data sets show no significant difference to the self-similarity values discussed before. Basically these values lie between the appropriate values of the self-similarity for learning and those of the self-similarity for prediction.
The graphs in figure 3.27 show similarity distributions for all pairs of peptide sets for a given feature set. Solid lines correspond to self-similarity between peptides of the learning set, while dashed lines represent self-similarity between peptides of the prediction set. From these histograms the consistency and characteristics of the data sets from different CoEPrA tasks can be discussed. Most significant patterns are shown for the graphs representing task CoEPrA 1 and 2. Here the curves of the different pairs of peptide sets, binding or non-binding are easy to discriminate. This supports the clues that for these tasks binding data should be easier to separate from non-binding data, which should support the prediction. For both other tasks of CoEPrA 3 and 4 little differentiation between binding and non-binding data is visible.

Is it possible to use similarity data to discriminate good from bad feature sets obtained from the GA? In the following figure 3.28 similarity between peptide sets is displayed for different typical individuals of the CoEPrA-2 task. To judge between good and bad individuals only curves from those data are shown, which do not require knowledge of prediction class association. Thus, the prediction data set is not divided into classes, but considered as a whole. Similarity is calculated between L+ P*, L- P* and L+ L-. Other combinations of intrasimilarity of the peptides from the learning data sets are not shown for improved clarity.

Though it was expected that these curves show clear pattern to discriminate the prediction quality of individuals, it turned out that for every assumed characteristic a counter part was found possesing the opposite prediction quality. If it is assumed i.e. that a peak in the histograms is characteristic for a good predicting individual other individuals can be found, showing the same characteristic peak, while possessing a weak prediction performance. It is even more difficult to find characteristics, which apply for all different CoEPrA tasks. The overall assumption is that the similarity measure is not a sufficient criterion to filter out good from bad individuals.

3.5.3 PCA applied to selected individuals to improve the prediction

In the previous section 3.5.1 different indicator quantities were used to rank individuals and exclude malicious individuals from the ranking. Here a method is described, which first ranks individuals to filter out one single individual from the entire final generation. Finally a principle component analysis is applied to improve the prediction rate. The following protocol was used to achieve the most promissing individual for PCA:

1. Sort all 200 individuals according to the value of min[MCC(aT)], the minimum value from all 4*20 test predictions, in a descending manner
2. Select only the 20 best individuals according to the previous ranking. Calculate for all 20 individuals the difference of MCC(aL) - min[MCC(aT)].
3. The individual with the smallest difference value is selected for PCA. In case that several individuals share the smallest difference value the one with the lowest variance var[MCC(aT)] is selected.
4. Apply the PCA to the selected individual

The minimum MCC of the test prediction seems to be a good measure for the preselection of individuals, because in many cases of learning by heart fluctuations in the test predictions

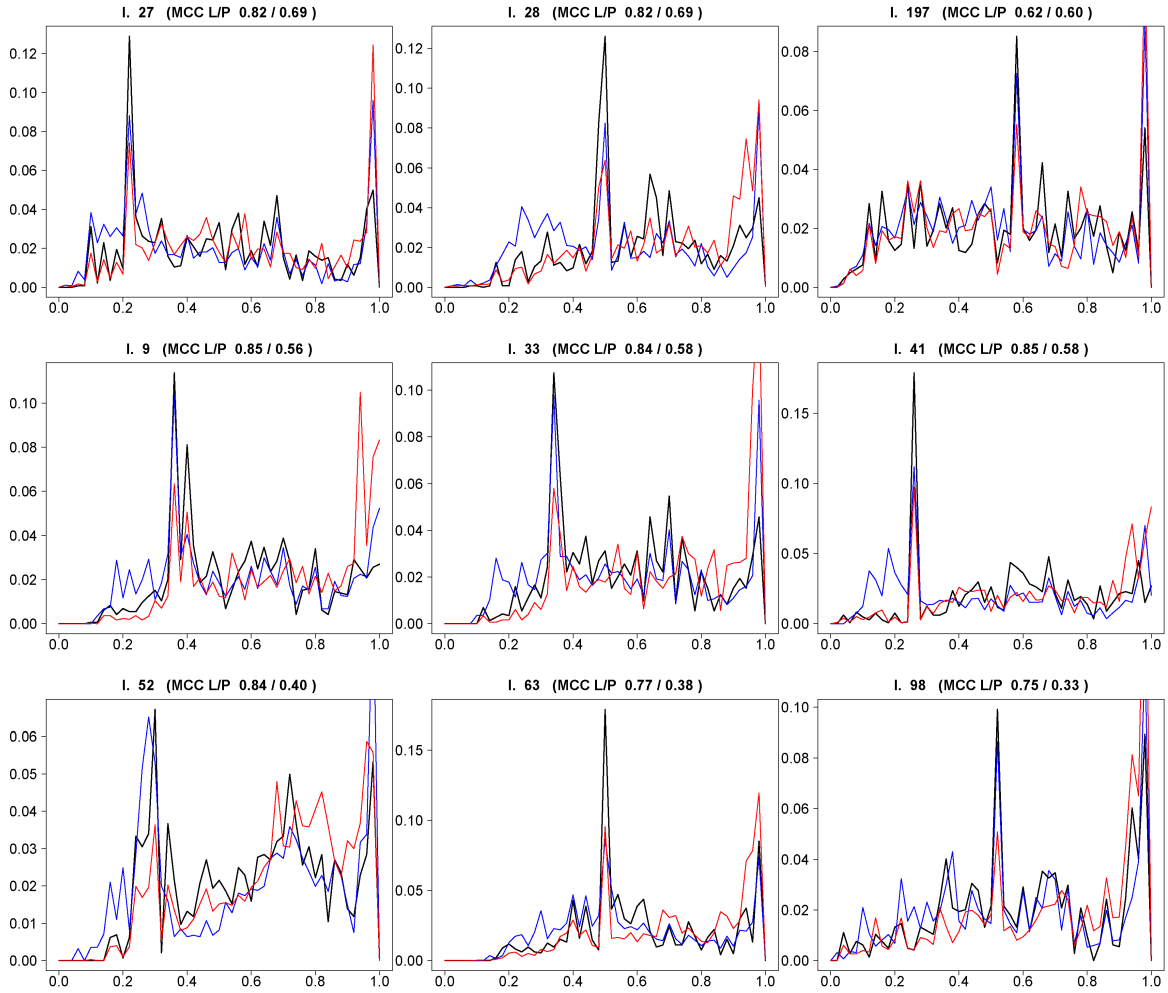


Figure 3.28: Similarity of pairs of peptide sets of different individuals considering the CoEPrA task 2. To discriminate individuals with good prediction performance (MCC P) from other individuals the pattern of curves should show typical characteristics. Color code used: black curves: L+L-, blue curves: L+P* and red curves: L-P*. P* marks the use of the complete prediction set. Other combinations are not shown. First row: individuals with good prediction performance and little learning by heart. Second row: individuals showing average prediction performance with more significant learning by heart. Third row: individuals with significant learning by heart and poor prediction performance.

performance occur. This is due to the variation of peptides in test prediction and test learning set. Another indicator of learning by heart is a high difference value between the average MCC of the test learning MCC(aL) and the minimum value of the test prediction. The PCA should finally overcome artificial constraints defined by the manual choice of the number of features per feature set in the genetic algorithm. In the case of the CoEPrA competition the complete number of features per feature set range between 6 to 9. Higher and lower number of features have been tested, but best results are achieved within this margin. The obtained eigenvalues are analyzed and those components assigned to the lowest eigenvalues are removed before solving the equation system. The following table 3.32 displays the first individuals of CoEPrA task 1 ranked according to the protocol described above.

# **indiv.**	$\frac{3}{4}$ **sets** **aL**	$\frac{1}{4}$ **sets** **aT**	**var<aT>**	**min<aT>**	**aL - min<aT>**	**tL**	**tP**
4	0.90	0.87	0.084	0.730	0.17	0.890	0.614
31	0.90	0.85	0.085	0.730	0.17	0.911	0.659
91	0.81	0.77	0.097	0.636	0.17	0.789	0.730
13	0.91	0.87	0.077	0.730	0.18	0.911	0.659
67	0.82	0.79	0.086	0.636	0.18	0.843	0.683

Table 3.32: Five ranked individuals from the last generation of the GA to select individual suitable for PCA considering CoEPrA task 1. Individuals consist of seven features: 2 F^+, 2 F^- and 3 F^0.Top ranked individual no. 4 (red) is selected for PCA. Eigenvalues for this individual are given in the text below. Test learning (aL) and test prediction (aT) are the average MCCs over 4*20 set variations. Random seed of 72, $\lambda_w = 10^{-10}$ and $w^+ = 0.50$ were used as parameters

According to the protocol individual no. 4 is ranked best in CoEPrA task 1 using $2F^+/2F^+/3F^0 = 7$ features. The PCA yields 7 eigenvalues and appropriate eigenvectors. The eigenvalues are sorted in ascending order. Smallest eigenvalues are associated with components, which are rather unspecific in their correlation such that these eigenvalues and their components are removed. For individual 4 the assigned eigenvalues are:

$$E_v = \begin{matrix} 0.00033 \\ 0.00558 \\ 0.01914 \\ 0.06356 \\ 0.17677 \\ 5.15066 \\ 13.3934 \end{matrix} \quad \text{remove smallest eigenvalues} \rightarrow \quad \begin{matrix} \sout{0.00033} \\ \sout{0.00558} \\ 0.01914 \\ 0.06356 \\ 0.17677 \\ 5.15066 \\ 13.3934 \end{matrix}$$

As to be expected from this parameter reduction the MCC value for recognition of the complete learning set is decreasing from 0.890 to 0.798. At the same time the MCC of the true prediction is increasing from 0.614 to 0.774. This is a dramatic performance increase yielding a better prediction result compared to the first ranked CoEPrA competitor in classification task 1 of 0.730. Based on this protocol the procedure was applied to all 4 CoEPrA tasks. The outcome is shown in the following table 3.33. Parameters used for the different CoEPrA tasks are as follows: For all tasks a weighting of w^+ as 0.50 was used. For the task 1 and 2 the lambda term was set to 10^{-10}, for CoEPrA 3 and 4 lambda was set to 10^{-9}. Feature sets for task 1 and 3 are composed

CoEPrA task	eigen values	removed EVs	before PCA MCC(tL)	before PCA MCC(tP)	after PCA MCC(tL)	after PCA MCC(tP)	MCC(aT) best competitor
1	0.00033 0.00558 0.01914 0.06356 0.17677 5.15066 13.3934	2	0.890	0.614	0.798	0.774	0.730
2	0.0000004 0.0000431 0.0069881 17.0306320 37.7404300 66.7015540	3	0.818	0.611	0.712	0.212	0.711
3	0.01557 0.04990 0.08816 1.09861 2.21814 3.77236 73.4830	1	0.686	0.250	0.490	0.310	0.356
4	0.00023 0.00039 0.00074 0.02262 0.06831	2	0.709	0.063	0.528	0.220	0.397

Table 3.33: PCA for selected individuals of all 4 CoEPrA classification contests. Change in learning and prediction results is shown with MCC values and compared to the first ranked competitors of the appropriate classification contest. The total number of features used per individual during the GA is equivalent to the number of eigenvalues listed. The selected individual is not in every classification task the one showing the best MCC in prediction. The MCC prediction value can improve after PCA steps but don't necessarily have to improve. For task 2 results after PCA are worsen, in task 4 the results improve after PCA, but nevertheless do not reach the quality of the best individual available in the present generation.

of 2 F^+, 2 F^- and 3 F^0 features, for task 2 2 F^+, 2 F^- and 2 F^0 features were used and for task 4 1 F^+, 2 F^- and 2 F^0 features were used. Random seeds for the different tasks are 72 for task 1, 48 for task 2, 7 for task 3 and 27 for task 4.

The results obtained after using the PCA based parameter reduction shows in 3 of 4 cases improvements in prediction performance. The protocol was applied strictly to obtain the individual of choice from each final GA generation. The selection of the omitted eigenvalues is based on the magnitude of the different eigenvalues obtained. There is some arbitrariness in deciding how many eigenvalues should be dismissed, but this has no influence of the general trend of the results. For CoEPrA task 2 the PCA method fails, delivering a result worse than the one before. In cases like CoEPrA 4 the result obtained after PCA is improving but still much worse compared to the best individual of the final GA generation selected. This demonstrates that until now, no reliable method was established to select best performing individuals from the appropriate generation. The selection scheme sometimes even favors individuals, which are quite apart from the quality

obtained from top performing individuals. The learning by heart, which is usually preventing the selection scheme to rank top individuals first can be handled with the PCA approach. With this approach chances raise to achieve good prediction performance at the end. In some cases the approach fails and prediction results are decreasing.

3.5.4 Discussion and summary of section 3.5

- Post-processing of the output from the GA is crucial to distinguish between good performing feature sets and those feature sets, which do extensively learning by heart.
- Filter approaches based on indicator values derived from test learning and test prediction performance often fail.
- Similarity measures between data sets used for learning and prediction are supposed to describe problems leading to learning by heart. Peptide set similarity was estimated to be a good indicator to detect individuals performing of strong learning by heart. In this study no good agreement between the similarity measure of individuals and their learning by heart behavior could be found.
- PCA derived parameter reduction after selecting one specific individual of the GA is in some cases improving the prediction performance. In some cases single individuals from the GA are performing better than the improved PCA treated individuals of the specific selection. One reason is that the individual selection protocol often selects individuals with an average prediction performance.
- A reliable way to select good individuals of the last GA generation for all different CoEPrA classification tasks is missing.

A number of techniques to separate good from bad performing individuals were analyzed. It was expected to find an approach, which works reliable for all four CoEPrA classification tasks. Instead it turned out that all studied methods could not ensure that bad performing individuals are eliminated in every case. Analyzing peptide similarity of learning and prediction sets for the four CoEPrA tasks can yield some understanding for the characteristics of the separate tasks but did not provide information on how to filter out bad performing individuals. Peptide self-similarity pattern for given feature sets for learning and prediction sets were searched to find repeating patterns. No pattern found was exclusively correlated to the prediction performance of individuals. The approach of first applying defined filter rules and finally using PCA to select eigenvalues, delivered promissing results for three out of the four CoEPrA tasks. For CoEPrA task 1,3 and 4 the prediction performance for the selected individual increased after the eigenvalue selection. Unfortunately the individuals selected for the PCA approach did not always belong to the top performing individuals of the appropriate final generation of the GA. In the case of CoEPrA task 4, the selected individual is a poor performing individual, which improves its' performance after applying the PCA in order to reach a prediction performance, which is already established by better performing individuals of the same final generation.
The conclusion of this section is that it was not possible to improve the selection of individuals with the studied methods. The GA provides generations of individuals, which are enriched by good, sometimes by excellent performing individuals. Nevertheless, learning by heart is difficult to detect and can lead to the selection of individuals, which have a high recognition performance but a weak prediction performance. The small peptide learning sets as they are provided by

CoEPrA can be one reason, why the feature selection of the GA is sometimes not performing well. Simple but more robust methods like a λ_w regularization optimization can be more effective in such cases.

Chapter 4

Summary and Outlook

In this work algorithms are presented, which can be used for knowledge based classification of data. Correlations of descriptors to target values are automatically derived by calculating a coefficient weighting matrix. The basic idea was to find an effective method designed for peptide binding prediction of specific MHC allels. The developed scoring function is able to handle any type of molecules, which can be characterized by feature based descriptors.
Application of the bare scoring function to evaluate HLA-A0201 binding peptides demonstrate how powerful this method is. ROC plots of recognition and prediction data demonstrate a robust method, which compares favorably with other well established classification approaches like support vector machine. Examination of MHC and MHC/TCR crystal structures give further inside about the binding of the antigen peptide with respect to MHC and TCR. Steric interactions and the peptide backbone play an essential role in binding to MHC in case of HLA A0201. The TCR interactions focus on the central residues of the peptide, which are not as tightly bound in the binding groove of the A*0201 α chain. Both ends of the peptide are important for interacting with the MHC. In position 2 and 9 anchor residues of the nonapeptide are present, which contain the fairly conserved residues leucine or valine. This study suggest that N- and C- terminal near residues are most important for binding of the peptide. Indeed recognition and prediction for peptides from the A0201 data set works fine if the central residue positions 4,5,6 are missing. The λ_w term is an effective but simple way to avoid learning by heart. In an unspecific way parameters are suppressed, which allow the usage of large feature vectors even for small data sets. The weighting parameter w^+ is influencing the contribution of the positive (+) class with respect to the negative (-) class and may be helpful to handle appropriatly asymmetric data sets. The CoEPrA competition allows direct comparison of the own results for different classification tasks with state-of-the-art research operating with a spectrum of different classification techniques. Without further optimization the achieved results rank in the top middle field of all competitors, sometimes even above. First results were obtained from sequence derived features and the use of different methods to prevent overfitting. The PCA to suppress less meaningful parameters or the lambda regularization are methods of choice.
Due to the success of feature selection in parameter regularization used by competitors in the CoEPrA contest, a framework for feature selection based on the scoring function was developed. The number of features to be selected is predefined. The genetic algorithm performs a heuristic search in the feature space to combine good performing features yielding optimal behaving feature sets. For the example of CoEPrA classification task 1 it was shown that the GA is converging towards an high level of true prediction performance regarding the average individuals of each generation. After 100 to 200 GA cycles the average individuals prediction performance has reached a plateau value. It could be shown that the GA is able to generate individuals, which

perform as good as the best competitors in the appropriate CoEPrA classification task. Unfortunately, almost every generation of individuals contains a number of bad or average performing individuals. Elimination of individuals, which are subject to learning by heart is difficult.
Similarities between learning and prediction data sets based on the selected features of each individual could not give a reliable hint, which individuals of the final generation are learning by heart. To avoid any bias in the selection of good individuals a protocol should be established, which allows to select successful individuals in all different CoEPrA tasks. The lack in a reliable individual selection method should be overcome by the approach of the PCA based parameter reduction. Small eigenvalues give a hint of unimportant parameters to be discarded. In three of four cases an improvement in the prediction rate could be detected. Nevertheless, it is not replacing the need of a more reliable individual filtering. Some individuals of the final generation are performing better than the individuals selected by the protocol even after the PCA treatment.
In conclusion it must be deduced that for some problems - like CoEPrA task 1 - feature selection works fine, but in other cases different approaches deliver better results. In these cases a simpler but more robust method is more reliable. This is the case for the lambda regularization, which can be optimized via boot strapping (currently evaluated by group member O. Demir).
One difficulty of the CoEPrA classification tasks are the small data sets available for learning. This has an impact on feature selection implemented via GA. For quality control the splitting of the learning set into test learning and test prediction set is required. This is limiting the number of data points available for learning even further.

4.1 Outlook

There are a number of possibilities to improve the scoring function approach. A soft-step function can be introduced using logarithmic or exponential based functions to evaluate, if a data point is assigned to one class type. Another idea is to use regression data to train the LSM, but to use the scoring function for a classification in recall and in prediction. This procedure is weighting data points with respect to their distance to the threshold used for the scoring function to separate classes.
Even without modifications of the scoring function one can improve prediction performance by using a bootstrapping method to optimize lambda regularization. This has been successfully implemented by a member of the Knapp work group, Oezguer Demir.
The scoring function can be applied for larger data sets derived from experimental binding studies, which are currently available. Meanwhile much more detailed information on binding and weak binding peptides for MHC allels is known. Based on measured IC50 or pIC50 values, classification of peptides can be established by introducing a threshold value, separating strong from weak binding peptides. This can be used to create new training data sets. The established method is capable of classifying large data sets with high accuracy. New data sets can improve prediction performance, if large training data sets are provided. Furthermore, drug molecules can be classified if pharmacophore fingerprints are used as descriptors.

Appendix A

Material

0	AAAKAAAAV	AAGIGIIQI	AAGIGILTV	AAPTPAAPA	ACDPHSGHF	ACRTVALTA	AFHHVAREL	AIAKAAAAV	AIMDKNIIL	AIVDKVPSV
10	ALAAVVTEV	ALACAAAAV	ALADAAAAV	ALADGVQKV	ALAKAAAAA	ALAKAAAAI	ALAKAAAAL	ALAKAAAAM	ALAKAAAAN	ALAKAAAAR
20	ALAKAAAAT	ALAKAAAAV	ALAKAAAEV	ALAKAAAFV	ALAKAAAGV	ALAKAAALV	ALAKAAAPV	ALAKAAEAV	ALAKAALAV	ALAKAANAV
30	ALAKAAPAV	ALAKAAYAV	ALAKAGAAV	ALAKAIAAV	ALAKAPAAV	ALAKARAAV	ALAKAYAAV	ALAKEAAAV	ALAKGAAAV	ALAKLAAAV
40	ALAKNAAAV	ALAKYAAAV	ALANGIEEV	ALAPAAAAV	ALASHLIEA	ALATAAAAV	ALAVAAAAV	ALCRWGLLL	ALEKAAAAV	ALFDGDPHL
50	ALFGALFLA	ALFKAAAAV	ALFPQLVIL	ALGLGLLPV	ALGRNSFEV	ALIHHNTHL	ALKKAAAAV	ALLNIKVKL	ALLPPINIL	ALMDKSLHV
60	ALMKAAAAV	ALMPLYACI	ALNELLQHV	ALNKMFCQL	ALNKMFYKL	ALNKMLCQL	ALQDSGLEV	ALQPGTALL	ALSDHHIYL	ALSDLEITL
70	ALSKAAAAV	ALSNLEVKL	ALSRKVAEL	ALSTGLIHL	ALWDIETGQ	ALWGFFPVL	ALWNLHGQA	ALYVDSLFF	AMAIHKQSQ	AMAKAAAAV
80	AMFQDPQER	ATAKAAAAV	AVAKAAAAV	AVFDRKSDA	AVGIGIAVV	AVVPFIVSV	AVVPFLVSV	CINGVCWTV	CLGGLITMV	CLGGLLTMV
90	CLTKWMILA	CLTSTVQLV	DLERKVESL	DLFGIWSKV	DLMGYIPLV	DLMLSPDDI	DLVHFASPL	DPKVKQWPL	DVASVIVTK	EAAGIGILT
100	ELIRVEGNL	ELTLGEFLK	ELVSEFSRM	ELVSEVSKV	EMFRELNEA	EVAPPLLFV	FAFRDLCIV	FIAGNSAYE	FIASNGVKL	FIDSYICQV
110	FIYAGSLSA	FKNIVTPRT	FLAKAAAAV	FLCKQYLNL	FLDEFMEGV	FLDGNELTL	FLDGNEMTL	FLDQVPFSV	FLEPGPVTA	FLFDGSPTY
120	FLGAAGSTM	FLGENISNF	FLGGTPVCL	FLGGTTVCL	FLKEPVHGV	FLLDKKIGV	FLLLADARV	FLLPSFAPD	FLLSLGIHL	FLLTRILTI
130	FLLWATAEA	FLPSDFFPS	FLQSRPEPT	FLTPKKLQC	FLWAIMHTE	FLWEFPHDL	FLWGPRALV	FLWGPRAYA	FLWTLEGDV	FLYCYFALV
140	FLYEAVPQL	FLYERVPQL	FLYGALLLA	FMFDLAAEL	FMFESPWNV	FMLDWFPTI	FTDQVPFSV	GAGIGVAVL	GAGIGVLTA	GELGFVFTL
150	GIAGGLALL	GIGIGVLAA	GIGILTVIL	GILGFVFTL	GILGFVFTM	GILGFVFTV	GILTVILGV	GIVPFIVSV	GIVPFLVSV	GLAPPQHEI
160	GLAPPQHLI	GLCTLVAML	GLDVLTAKV	GLHCYEQLV	GLIEKNIEL	GLIMVLSFL	GLLGFVFTL	GLLGNVSTV	GLLGTLVQL	GLLGWSPQA
170	GLPVEYLQV	GLQDCTMLV	GLRDLAVAV	GLSEFTEYL	GLSPTVWLS	GLSRYVARL	GLVPFIVSV	GLVPFLVSV	GLYDGMEHL	GLYPGLIWL
180	GLYSSTVPV	GMLGFVFTL	GMNCRPILT	GMNERPILT	GMNKRPILT	GMNRHPILT	GMNRRPILT	GQLGFVFTL	GTLGFVFTL	GTLGIVCPI
190	GTLSKIFKL	GVALQTMKQ	GVLGFVFTL	GVLVGVALI	HEIRVEGNL	HLEGKVILV	HLESLFTAV	HLGNVKYLV	HLIDYLVTS	HLIKVEGNL
200	HLIRVEGNL	HLLVGSSGL	HLSLRGLPV	HLSTAFARV	HLYQGCQVV	HLYSHPIIL	HMTEVVRHC	HMTEVVRRC	IAGIGILAI	IIDQVPFSV
210	IISAVVGIL	IISCTCPTV	IISLWDQSL	ILAGYGAGV	ILAKFLHWL	ILAPPVVKL	ILAQVPFSV	ILDQKINEV	ILDQVPFSV	ILDTGTIQL
220	ILFEPVHGV	ILFGHENRV	ILGFVFTLT	ILHNGAYSL	ILKEPVHGV	ILKEYVHGV	ILKSPVHGV	ILLLCLIFL	ILMEHIHKL	ILMQVPFSV
230	ILSPFMPLL	ILSPLTKGI	ILSQVPFSV	ILTVILGVL	ILWEPVHGV	ILYEPVHGV	IMDKNIILK	IMDQVPFSV	IMIGVLVGV	ITAQVPFSV
240	ITDQVPFSV	ITFQVPFSV	ITMQVPFSV	ITSQVPFSV	ITWQVPFSV	ITYQVPFSV	IVGAETFYV	KACDPHSGH	KARDPHSGH	KASEKIFYV
250	KIFGSLAFL	KILSVFFLA	KINEPVIII	KINEPVIIL	KINEPVILI	KINEPVILL	KINEPVLII	KINEPVLIL	KINEPVLLI	KINEPVLLL
260	KKREEAPSL	KLAEYVAKV	KLAKAAAAV	KLFCQLAKT	KLGEFYNQM	KLHLYSHPI	KLIANNTRV	KLLEPVLLL	KLLPENNVL	KLNEILWSI
270	KLNEPVIII	KLNEPVIIL	KLNEPVILI	KLNEPVILL	KLNEPVLII	KLNEPVLIL	KLNEPVLLI	KLNEPVLLL	KLPAQFYIL	KLPQLCTEL
280	KLTPLCVTL	KLTSLCNTV	KLVALGINA	KLVANNTRL	KMFCQLAKT	KMFYQLAKT	KMVELVHFL	KTWGQYWQV	KVAELVHFL	KVLEYVIKV
290	KYLATASTM	LAGIGLIAA	LERPGGNEI	LIVIGILIL	LLAQFTSAI	LLARNSFEV	LLCLIFLLV	LLDFVRFMG	LLDGTATLR	LLDVPTAAV
300	LLFAGVQCQ	LLFDRPMHV	LLFGYPVYV	LLGANSFEV	LLGATCMFV	LLGRASFEV	LLGRDSFEV	LLGRNAFEV	LLGRNSAEV	LLGRNSEEM
310	LLGRNSFAV	LLGRNSFEM	LLGRNSFEV	LLGRRSFEV	LLIENVASL	LLLCLIFLL	LLLLTVLTV	LLMDCSGSI	LLMGTLGIV	LLNATAIAV
320	LLNATDIAV	LLPENNVLS	LLQYWSQEL	LLSSNLSWL	LLSVPLLLG	LLTEVETYV	LLWAARPRL	LLWFHISCL	LLWKGEGAV	LLWTLVVLL
330	LLYDWDFGL	LMAQEALAF	LMIIPLINV	LMWAKIGPV	LQTTIHDII	LTVILGVLL	LVVLGLLAV	MDHARHGFL	MIMVKCWMI	MLDLQPETT
340	MLGTHTMEV	MLLALLYCL	MLLAVLYCL	MLLSVPLLL	MLMAQEALA	MLWEGFTYI	MMQDIDFYL	MMRKLAILS	MMWYWGPSL	MVDGTLLLL
350	MVDGTTLLL	NLGPWIQQV	NLLPKLHIV	NLQSLTNLL	NLSWLSLDV	NLTISDVSV	NLVPMVATV	NMFCQLAKT	NMFTPYIGV	PILTIITLE
360	PLDGEYFTL	PLKQHFQIV	PLLPIFFCL	PLQPEQLQV	PLSSSVPSQ	PLTFGWCYK	PLTSIISAV	QAGIGILLA	QIRGRERFE	QLAKTCPVQ
370	QLFHLCLII	QLIDKVWQL	QLQARILAV	QLSLLMWIT	QMFCQLAKT	QMVTTTNPL	QVCERIPTI	RGPGRAFVT	RIIPRHLQL	RILGAVAKV
380	RLCVQSTHV	RLDSYVRSL	RLGFLHSGT	RLGRNSFEV	RLIRVEGNL	RLLDYVVNI	RLLQETELV	RLMKQDFSV	RLNMFTPYI	RLPKDFRIL
390	RLPRIFCSC	RLSSNSRIL	RLTRFLSRV	RLVTLKDIV	RMFPNAPYL	RMGAVTTEV	RMPEAAPPV	RMYSPISIL	RTLDKVLEV	RTQDENPVV
400	RVIEVLQRA	SAHKGFKGV	SIIVRALEV	SILVRALEV	SIPSGGIGV	SIPSGGLGV	SLADTNSLA	SLAGGIIGV	SLDDYNHLV	SLDQSVVEL
410	SLFEGIDFY	SLFNTVATL	SLFPGKLEV	SLGGLLTMV	SLHVGTQCA	SLIGHLQTL	SLINVGLIS	SLIVRALEV	SLKKNSRSL	SLLAPGAKQ
420	SLLGGDVVS	SLLGLLVEV	SLLLELEEV	SLLMWITQC	SLLPAIVEL	SLLPPDALV	SLLPPTALV	SLLQHLIGL	SLLVRALEV	SLMAFTAAV
430	SLPDFGISY	SLPSGGIGV	SLPSGGLGV	SLRELGSGL	SLSEKTVLL	SLSRFSWGA	SLVIVTTFV	SLWGQPAEA	SLYADSPSV	SLYAVSPSV
440	SLYGGTTTI	SLYNTIAVL	SLYNTVATL	SLYSFPEPE	SMVGNWAKV	STAPPAHGV	STAPPHVNV	STNRQSGRQ	STPPPGTRV	SVASTITGV
450	SVFAGVVGV	SVRDRLARL	SVYDFFVWL	TGAPVTYST	TITDQVPFS	TIWVDPYEV	TLDDLIAAV	TLDSQVMSL	TLEEITGYL	TLFIGSHVV
460	TLGIVCPIC	TLHEYMLDL	TLIEDILGV	TLIKIQHTL	TLNAWVKVV	TLSKIFKLG	TLSPGKNGV	TLWVDPYEV	TMDHARHGF	TTAEEAAGI
470	TVILGVLLL	VDGIGILTI	VILGVLLLI	VISNDVCAQ	VIYQYMDDL	VKTDGNPPE	VLAGLLGNV	VLAKAAAAV	VLATLVLLL	VLDGLDVLL
480	VLEETSVML	VLEWRFDSR	VLFRGGPRG	VLFSSDFRI	VLHDDLLEA	VLLCESTAV	VLLDYQGML	VLPDVFIRC	VLQAGFFLL	VLQWASLAV
490	VLSPLPSQA	VLVKSPNHV	VLYRYGSFS	VMAGVGSPY	VMAPRTLVL	VMNILLQYV	VVHFFKNIV	VVLGVVFGI	VYDGREHTV	WILRGTSFV
500	WLDQVPFSV	WLEPGPVTA	WLNEILWSI	WLSLLVPFV	WLWYIKIFI	WTDQVPFSV	YIGEVLVSV	YLATASTMD	YLDNGVVFV	YLDPAQQNL
510	YLDQVPFSV	YLEPGPVTA	YLEPGPVTI	YLEPGPVTL	YLEPGPVTV	YLGEVIVSV	YLGEVLVSV	YLKEPVHGV	YLKKIKNSL	YLKKIQNSL
520	YLKTIQNSL	YLLEMLWRL	YLLPAIVEL	YLLPAIVHI	YLLPRRGPR	YLNKIQNSL	YLQLVFGIE	YLSGANLNL	YLVAYQATV	YLVSFGVWI
530	YLVTRHADV	YMDDVVLGA	YMDGTMSQV	YMLDLQPET	YMNGTMSQV	YTAFTIPSI	YTDQVPFSV	YVDPVITSI		

Table A.1: All 538 binding sequences for HLA-A*0201 used for the set $\mathbb{S}^+$

0	7ZNF	1AGQ	1BRX	1C51	1BCC	1EPW	1A75	1E4T	1AQU	1O23
10	6RLX	1AIR	8TIM	1BQP	1P3H	1E1H	1LF4	1E3E	1ALB	1NMM
20	6Q21	1AFO	1A0R	1EIS	1UJL	1DXR	1E0C	1DX1	1AI1	1NCI
30	1HNE	3MRA	1A12	1C01	1GWY	1GWC	1HRK	1DSV	1AFV	1NAS
40	1EAD	1AUN	3ZNC	1C4R	1RK4	1G6R	1RIE	1DJ2	1A2Y	1N9P
50	1VMO	1AUV	1A38	1GGX	1QKK	1FYT	1UOY	1DF3	1A1H	1MNU
60	821P	1A04	2SQC	1FHF	1B9C	1L0X	1H1V	1DD7	1914	1MBY
70	1BOM	1AF6	1BUG	1H4Y	1BFA	1ITZ	1GL5	1CQZ	1R2A	1MBE
80	1AHL	1A06	1BYO	1BPO	1BD2	1IR1	1GL2	1CL7	1QLX	1M4M
90	1SRA	1AXM	7PCK	1AB1	1AO7	1LFJ	1G74	1CE6	1PA2	1M3V
100	1DOX	1AZD	1BYY	1H8P	1A2X	1OM0	1FWU	1CDK	1P8J	1LB1
110	1MSP	1AIW	2VSG	1GRW	1A2C	1OED	1FRB	1C2B	1ORS	1KCM
120	1FAT	1A0D	1B10	1JV1	1D9K	1TCR	1FKW	1BLN	1OMX	1KBQ
130	1BGK	1BB9	1C3A	1O7N	1CNE	1QF3	1F93	1BKX	1OKQ	1K2F
140	7UPJ	1A05	1EG5	1H0H	1CJK	1PJU	1F81	1BGX	1OGP	1JJO
150	6UPJ	1BA1	1EHD	1B8M	1FV3	1WGT	1EDH	1BBS	1OCP	1JI9
160	5UPJ	1BKD	1BQF	1GDJ	1EZF	1BR1	1E4W	1AX8	1OAA	1IWE
170	1ISN	1IQ1	1IKN	1IG3	1IFQ	1IFA	1IAL	1I7W	1I7E	1I6Z
180	1I07	1HQV	1HQ8	1HN3	1H96	2ZNC	2MSS	2IAD	2DLF	1SUH
190	1HA7	1GK8	1BX7	1BWK	1BK6	1BJT	1ASZ	1AI9	1A6R	1A4H
200	1A48	1A2V								

Table A.2: Pdb code of proteins' whose sequences were concatenated to generate 10,000 random nonapeptides used as A0201 nonbinders for the set $\mathbb{S}^-$

A.0.1 Position dependent amino acid distribution in the set of binding HLA-A0201 peptides

aa	Pos.1	Pos.2	Pos.3	Pos.4	Pos.5	Pos.6	Pos.7	Pos.8	Pos.9
A	16.17	2.79	14.68	5.95	11.15	11.90	16.54	11.34	5.95
C	0.93	0.37	1.30	2.04	0.56	2.04	1.49	1.67	1.30
D	1.30	0.37	7.25	7.81	2.23	2.23	1.12	1.49	0.37
E	1.30	0.56	2.42	12.08	2.97	3.53	2.23	8.74	1.30
F	7.62	0.19	6.32	1.30	6.13	4.83	10.59	3.53	0.93
G	8.74	0.37	8.18	11.52	11.34	4.83	3.90	7.43	0.74
H	2.60	0.00	1.86	0.93	2.60	1.86	4.46	2.42	0.37
I	7.25	8.92	4.28	3.90	4.46	8.74	8.36	5.76	8.36
K	8.18	0.56	2.23	11.71	3.53	2.23	0.74	4.09	0.74
L	8.55	67.47	12.08	3.35	8.36	10.59	12.08	12.08	27.70
M	2.60	8.18	2.79	0.56	0.93	1.86	1.30	1.67	1.49
N	1.49	0.00	6.88	2.23	5.02	2.23	3.35	2.79	0.19
P	1.49	0.19	3.16	10.04	8.36	8.92	5.95	3.53	0.00
Q	1.86	0.37	2.42	5.58	3.35	2.23	3.16	4.09	1.67
R	4.46	0.00	1.67	5.76	5.02	1.30	1.49	3.53	0.93
S	9.67	0.00	6.13	6.32	2.04	6.69	5.20	10.78	1.67
T	3.35	4.46	3.16	3.90	4.83	5.95	4.65	8.74	3.90
V	5.20	4.83	4.65	3.53	11.15	15.99	9.29	2.60	41.64
W	1.30	0.00	4.09	0.74	2.04	0.56	2.23	0.93	0.00
Y	5.95	0.37	4.46	0.74	3.90	1.49	1.86	2.79	0.74
	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

Table A.3: Position dependent amino acid distribution of the HLA-A0201 binding set $\mathbb{S}^+$. All numbers are percentage of occurence.

A.1 Features used for CoEPrA

In the CoEPrA 2006 [5] contest each residue position of the sequences is described by 643 features for classification tasks 1-4. These 643 features, which are derived from physicochemical properties, are unique descriptors for the type of amino acid, but are independent of the position where they occur. Most of the features of CoEPrA 2006 were taken from the public accessable AAindex database [32], which provides a number of amino acid type specific indices published in literatur. The database currently (Sept. 2008) contains 544 type specific indices in the AAindex1 listing. Due to the organizers of CoEPrA 2006 no global descriptors are used as features. Furthermore the last 20 features of each residue type specific descriptor are represented by Miyazawa-Jernigan indices [61].
Unfortunately the organizers of CoEPrA 2006 did not provide more detailed information about the origin or characteristics of the features on request. Thus there are 79 of those 643 involved features of unknown derivation.

A.2 CoEPrA classification datasets without featurevectors

The classes assigned to the peptides are marked by -1 or +1 for non-binding or binding peptides for the learning set. In the prediction set the 1 is replaced by an "x", yielding a -x for non-binding and +x for binding peptides of the prediction set.

learning set				prediction set			
binding set		**nonbinding set**		**binding set**		**nonbinding set**	
molecules	class	molecules	class	molecules	class	molecules	class
IYDPFPVTV	1	ILDPFPVTD	-1	FLDDHFCTV	+x	ALCRWGLLL	-x
YLSPGPVTA	1	ILDPFPVTY	-1	FLFPGPVTA	+x	ALMPLYACI	-x
LLFGYPVYV	1	ILDPFPVTH	-1	FLFPLPPEV	+x	ALPYWNFAT	-x
YLFDGPVTA	1	SLHVGTQCA	-1	FLKPFYHNV	+x	FLLSLGIHL	-x
ILDPFPVTT	1	HLLVGSSGL	-1	FLWPIYHDV	+x	FLLTRILTI	-x
RLWPLYPNV	1	NLQSLTNLL	-1	HLYSHPIIL	+x	FVTWHRYHL	-x
YLFPGPVWA	1	SLNFMGYVI	-1	ICDPFPVTV	+x	GILTVILGV	-x
YAIDLPVSV	1	ITSQVPFSV	-1	ILDDFPVTV	+x	GLGQVPLIV	-x
YLFNGPVTV	1	VCMTVDSLV	-1	ILDDLPPTV	+x	GLSRYVARL	-x
ILDPFPVTF	1	LLMGTLGIV	-1	ILDPFPPEV	+x	HLESLFTAV	-x
YLWPGPVTV	1	ALIHHNTHL	-1	ILDPFPPTV	+x	IDDPFPVTV	-x
RLWPFYHNV	1	MLDLQPETT	-1	ILDPFPVTC	+x	IGDPFPVTV	-x
YLAPGPVTA	1	YVITTQHWL	-1	ILDPFPVTI	+x	ILDPFPVTE	-x
IADPFPVTV	1	ITFQVPFSV	-1	ILDPFPVTL	+x	ILDPFPVTN	-x
YLYPGPVTA	1	KTWGQYWQV	-1	ILDPFPVTM	+x	ILDPFPVTQ	-x
YLFPGPETA	1	ITDQVPFSV	-1	ILDPIPPTV	+x	ILKPLYHNV	-x
ILDPFPVTP	1	LLAQFTSAI	-1	ILKEPVHGV	+x	INDPFPVTV	-x
FLWPFYPNV	1	VLHSFTDAI	-1	ILNPFYHNV	+x	ITAQVPFSV	-x
FLDQVPFSV	1	ILDPFPVTK	-1	ILWPIYHNV	+x	ITWQVPFSV	-x
FLWPFYHNV	1	YMNGTMSQV	-1	ILWQVPFSV	+x	IWDPFPVTV	-x
ILWPLFHEV	1	ILDPFPVTW	-1	IMDQVPFSV	+x	KIFGSLAFL	-x
ILWPLYPNV	1	FTDQVPFSV	-1	IQDPFPVTV	+x	KLPQLCTEL	-x
ILDQVPFSV	1	KLHLYSHPI	-1	ISDPFPVTV	+x	LLWFHISCL	-x
ILNPFYPDV	1	ILDPFPVTS	-1	ITDPFPVTV	+x	LMAVVLASL	-x
FLWPLYPNV	1	YTDQVPFSV	-1	IVDPFPVTV	+x	LQTTIHDII	-x
FLNPFYPNV	1	IFDPFPVTV	-1	NMVPFFPPV	+x	MLGTHTMEV	-x
FLNPIYHDV	1	CLTSTVQLV	-1	RLWPIYHDV	+x	NLSWLSLDV	-x
YLFPGTVTA	1	YLWQYIFSV	-1	RLWPIYHNV	+x	RLLQETELV	-x
YLCPGPVTA	1	IHDPFPVTV	-1	YLAPGPVTV	+x	RLNPFYHDV	-x
YLFPPPVTV	1	RLMKQDFSV	-1	YLEPGPVTL	+x	SIISAVVGI	-x
ILFPGPVTA	1	VMGTLVALV	-1	YLFNGPVTA	+x	SLDDYNHLV	-x
IIDPFPVTV	1	ILYQVPFSV	-1	YLFPCPVTA	+x	SLYADSPSV	-x
ILDPFPVTA	1	IPDPFPVTV	-1	YLFPDPVTA	+x	SVYDFFVWL	-x
FLWPIYHNV	1	GLLGWSPQA	-1	YLFPGPFTA	+x	TLGIVCPIC	-x
ILFPFVHSV	1	GLYSSTVPV	-1	YLFPGPFTV	+x	TLHEYMLDL	-x
ILDPFPVTG	1	IISCTCPTV	-1	YLFPGPMTA	+x	TTAEEAAGI	-x
YLFPFPITV	1	FLCKQYLNL	-1	YLFPGPMTV	+x	VLIQRNPQL	-x
ILFPFPVEV	1	YLFPGPVTG	-1	YLFPGPSTA	+x	VLLDYQGML	-x
ILDDFPPTV	1	GTLGIVCPI	-1	YLFPGPVQA	+x	VTWHRYHLL	-x
ILDPLPPTV	1	RLWPFYPNV	-1	YLFPGPVTA	+x	WILRGTSFV	-x
IMDPFPVTV	1	YLKPGPVTA	-1	YLFPGVVTA	+x	WLDQVPFSV	-x
ILDPFPPPV	1	YLMPGPVTA	-1	YLFPPPVTA	+x	YLFDGPVTV	-x
ILDPFPITV	1	YMLDLQPET	-1	YLNPGPVTA	+x	YLFQGPVTA	-x
ILDPFPVTV	1	PLLPIFFCL	-1	YLWDHFIEV	+x	YLWQYIPSV	-x
		RLNPLYPNV	-1				

Table A.4: Coepra 2006 classification data of the learning and prediction set for **problem 1**

learning set				prediction set			
binding set		**nonbinding set**		**binding set**		**nonbinding set**	
molecules	class	molecules	class	molecules	class	molecules	class
FDSTGNLI	1	FESTGNLD	-1	FEETGNLN	+x	AESKSVII	-x
FESTSNLI	1	FKSTGNLI	-1	FEITGNLN	+x	DESTGNLI	-x
FESTWNLI	1	FESTGNLR	-1	FEKTGNLN	+x	EESTGNLI	-x
FGSTGNLI	1	FFSTGNLI	-1	FEMTGNLN	+x	FAFPGELL	-x
FESTGWLI	1	FESTGNLQ	-1	FEPTGNLN	+x	FAFWAFVV	-x
FESTINLI	1	FESTGNLH	-1	FESGGNLI	+x	FASTGNLI	-x
FESDGNLI	1	FESTGNLG	-1	FESHGNLI	+x	FESTDNLI	-x
FESTLNLI	1	FISTGNLI	-1	FESLGNLI	+x	FESTENLI	-x
FESTVNLI	1	QTFVVGCI	-1	FESMGNLI	+x	FESTGNAI	-x
LEILNGEI	1	NEKSFKDI	-1	FESNGNLI	+x	FESTGNGI	-x
FESTGKLI	1	FQSTGNLI	-1	FESQGNLI	+x	FESTGNLK	-x
DGLGGKLV	1	FLSTGNLI	-1	FESSGNLI	+x	FESTGNLL	-x
FESEGNLI	1	FESTGNKI	-1	FESTFNLI	+x	FESTGNLN	-x
FESKGNLI	1	FESTGNLM	-1	FESTGALI	+x	FESTGNLP	-x
FEHTGNLN	1	FESTGNDI	-1	FESTGFLI	+x	FESTGNLT	-x
FESWGNLI	1	FESTGNLW	-1	FESTGGLI	+x	FESTGNLV	-x
FESTANLI	1	KESTGNLI	-1	FESTGHLI	+x	FESTGNLY	-x
FEFTGNLN	1	FESTGNPI	-1	FESTGILI	+x	FESTGNMI	-x
FESTGVLI	1	PESTGNLI	-1	FESTGLLI	+x	FESTGNSI	-x
FESAGNLI	1	FESTGNLA	-1	FESTGMLI	+x	FESTGNTI	-x
FESPGNLI	1	FESTGNNI	-1	FESTGNLF	+x	FESTGNVI	-x
FESTGNFI	1	FESTGNLS	-1	FESTGNRI	+x	FESTKNLI	-x
FESTGNLI	1	FESTGNEI	-1	FESTGNWI	+x	FESTRNLI	-x
FESFGNLI	1	VESTGNLI	-1	FESTGNYI	+x	FESTYNLI	-x
FESRGNLI	1	FESTGNII	-1	FESTGPLI	+x	FHSTGNLI	-x
FESYGNLI	1	FESTGELI	-1	FESTGQLI	+x	FLHPSMPV	-x
FESTPNLI	1	HESTGNLI	-1	FESTGSLI	+x	FMSTGNLI	-x
FEATGNLN	1	FESTGNQI	-1	FESTGTLI	+x	FNSTGNLI	-x
FEDTGNLN	1	AESTGNLI	-1	FESTGYLI	+x	FSSTGNLI	-x
FEQTGNLN	1	SESTGNLI	-1	FESTHNLI	+x	FTSTGNLI	-x
FESTGRLI	1	GESTGNLI	-1	FESTMNLI	+x	FVSTGNLI	-x
FENTGNLN	1	FESTGDLI	-1	FESTQNLI	+x	FWSTGNLI	-x
FESVGNLI	1	IESTGNLI	-1	FESTTNLI	+x	FYSTGNLI	-x
FESIGNLI	1	MESTGNLI	-1	FETTGNLN	+x	HAIHGLLV	-x
FEGTGNLN	1	QESTGNLI	-1	FEVTGNLN	+x	LESTGNLI	-x
FERTGNLN	1	NESTGNLI	-1	FEWTGNLN	+x	RESTGNLI	-x
FELTGNLN	1	WESTGNLI	-1	FEYTGNLN	+x	TESTGNLI	-x
		FESTGNHI	-1	FPSTGNLI	+x	YESTGNLI	-x
		FESTNNLI	-1				

Table A.5: Coepra 2006 classification data of the learning and prediction set for **problem 2**

learning set				prediction set			
binding set		**nonbinding set**		**binding set**		**nonbinding set**	
molecules	class	molecules	class	molecules	class	molecules	class
SVMDPLIYA	1	VVHFFKNIV	-1	ALAKAAAAM	+x	AAAKAAAAV	-x
VLLDYQGML	1	VCMTVDSLV	-1	ALLAGLVSL	+x	ALAKAAAAV	-x
LMIGTAAAV	1	LLGCAANWI	-1	ALMPLYACI	+x	ALCRWGLLL	-x
TVLRFVPPL	1	SAANDPIFV	-1	ALVLLMLPV	+x	ALIHHNTHL	-x
NLGNLNVSI	1	TTAEEAAGI	-1	ALYGALLLA	+x	ALLSDWLPA	-x
ILHNGAYSL	1	LTVILGVLL	-1	AMVGAVLTA	+x	ALSTGLIHL	-x
SIISAVVGI	1	LVSLLTFMI	-1	AVIGALLAV	+x	AMFQDPQER	-x
VLAKDGTEV	1	QMTFHLFIA	-1	DLMGYIPLV	+x	AMKADIQHV	-x
YLEPGPVTI	1	ALPYWNFAT	-1	FLLTRILTI	+x	AMLQDMAIL	-x
FLYNRPLSV	1	FVTWHRYHL	-1	FLYGALLAA	+x	AVAKAAAAV	-x
FLWGPRALV	1	SLNFMGYVI	-1	FLYGALLLA	+x	DPKVKQWPL	-x
ILDQVPFSV	1	GIGILTVIL	-1	FLYGALVLA	+x	FAFRDLCIV	-x
ILSSLGLPV	1	IVMGNGTLV	-1	FLYGGLLLA	+x	FLAGALLLA	-x
LLFLGVVFL	1	SLSRFSWGA	-1	FTDQVPFSV	+x	FLEPGPVTA	-x
YLVAYQATV	1	TVILGVLLL	-1	FVWLHYYSV	+x	FMGAGSKAV	-x
YLEPGPVTV	1	WTDQVPFSV	-1	GLLGNVSTV	+x	FVDYNFTIV	-x
ILSPFMPLL	1	AIAKAAAAV	-1	GLLGWSPQA	+x	FVNHDFTVV	-x
YLSPGPVTA	1	ITSQVPFSV	-1	GLQDCTMLV	+x	GLACHQLCA	-x
IIDQVPFSV	1	ALAKAAAAI	-1	GLSRYVARL	+x	GLCFFGVAL	-x
YMNGTMSQV	1	GLGQVPLIV	-1	GLYSSTVPV	+x	GLVDFVKHI	-x
FLCWGPFFL	1	LLSSNLSWL	-1	GLYYLTTEV	+x	GLYLSQIAV	-x
LLFRFMRPL	1	SIIDPLIYA	-1	HLYSHPIIL	+x	HLAVIGALL	-x
ITWQVPFSV	1	YLVTRHADV	-1	ILAQVPFSV	+x	HLESLFTAV	-x
LLAVLYCLL	1	LIGNESFAL	-1	ILMQVPFSV	+x	HLLVGSSGL	-x
GIRPYEILA	1	FLLPDAQSI	-1	ILSQVPFSV	+x	HLYQGCQVV	-x
GLFLTTEAV	1	CLALSDLLV	-1	ILYQVPFSV	+x	IISCTCPTV	-x
YTYKWETFL	1	LLGRNSFEV	-1	IMPGQEAGL	+x	ILAGYGAGV	-x
ALVGLFVLL	1	LLAVGATKV	-1	ITFQVPFSV	+x	ILDEAYVMA	-x
SLDDYNHLV	1	MLLAVLYCL	-1	ITMQVPFSV	+x	ILLSIARVV	-x
FLLRWEQEI	1	AIYHPQQFV	-1	ITYQVPFSV	+x	ILTVILGVL	-x
SLLPAIVEL	1	ALAKAAAAL	-1	IVGAETFYV	+x	ITAQVPFSV	-x
YLSPGPVTV	1	FVNHRFTVV	-1	KIFGSLAFL	+x	ITDQVPFSV	-x
GLIMVLSFL	1	WILRGTSFV	-1	KILSVFFLA	+x	KLAGGVAVI	-x
SLYADSPSV	1	TLDSQVMSL	-1	KTWGQYWQV	+x	LLACAVIHA	-x
RLLQETELV	1	GLYGAQYDV	-1	LLAQFTSAI	+x	LLPLGYPFV	-x
IMDQVPFSV	1	MLASTLTDA	-1	LLDVPTAAV	+x	LLSCLGCKI	-x
YLLPAIVHI	1	AIIDPLIYA	-1	LLFGYPVYV	+x	LLVFACSAV	-x
FLLLADARV	1	FLGGTPVCL	-1	LLLCLIFLL	+x	LLVVMGTLV	-x
ALMDKSLHV	1	LMLPGMNGI	-1	LLLLGLWGL	+x	LLWFHISCL	-x
YLYPGPVTA	1	RLMIGTAAA	-1	LLWQDPVPA	+x	LMAVVLASL	-x
HMWNFISGI	1	LLFLLLADA	-1	LLWSFQTSA	+x	LQTTIHDII	-x
YLAPGPVTV	1	GTLGIVCPI	-1	MALLRLPLV	+x	MLGNAPSVV	-x
MLGTHTMEV	1	KLFPEVIDL	-1	MMWYWGPSL	+x	NLQSLTNLL	-x
MTYAAPLFV	1	IAGGVMAVV	-1	NLYVSLLLL	+x	QLFHLCLII	-x
YLSQIAVLL	1	GLYRQWALA	-1	QLFEDNYAL	+x	QVMSLHNLV	-x
YLMPGPVTV	1	MLQDMAILT	-1	RLMKQDFSV	+x	RLLGSLNST	-x
WLDQVPFSV	1	VILGVLLLI	-1	RMFAANLGV	+x	RLTEELNTI	-x
SLYFGGICV	1	CLTSTVQLV	-1	RMYGVLPWI	+x	RLVSGLVGA	-x
YLLALRYLA	1	ILLLCLIFL	-1	SVYDFFVWL	+x	RMPAVTDLV	-x
SLLTFMIAA	1	DMWEHAFYL	-1	VLAGLLGNV	+x	SLADTNSLA	-x

table continued on the following page...

...**table continued from previous page.**

learning set				**prediction set**			
binding set		**nonbinding set**		**binding set**		**nonbinding set**	
molecules	class	molecules	class	molecules	class	molecules	class
GLMTAVYLV	1	ALTVVWLLV	-1	VLIQRNPQL	+x	SLHVGTQCA	-x
FLLSLGIHL	1	LLPSLFLLL	-1	VLLLDVTPL	+x	SVYVDAKLV	-x
FVVALIPLV	1	WMNRLIAFA	-1	VLLPSLFLL	+x	TLLVVMGTL	-x
YLWPGPVTV	1	PLLPIFFCL	-1	VLTALLAGL	+x	VALVGLFVL	-x
FLYGALRLA	1	ALAKAAAAA	-1	VMGTLVALV	+x	VIHAFQYVI	-x
LLLEAGALV	1	FLPWHRLFL	-1	VVLGVVFGI	+x	VLHSFTDAI	-x
YLFPGPVTV	1	SLAGFVRML	-1	WLSLLVPFV	+x	VLVGGVLAA	-x
ILFTFLHLA	1	TLGIVCPIC	-1	YAIDLPVSV	+x	VVMGTLVAL	-x
RLPLVLPAV	1	KLTPLCVTL	-1	YLAPGPVTA	+x	WLEPGPVTA	-x
YMDDVVLGV	1	LLCLIFLLV	-1	YLDLALMSV	+x	WLLIDTSNA	-x
GILTVILGV	1	RIWSWLLGA	-1	YLDQVPFSV	+x	YALTVVWLL	-x
NMVPFFPPV	1	SLLEIGEGV	-1	YLFPGPVTA	+x	YLEPGPVTL	-x
FLYGAALLA	1	RLLDDTPEV	-1	YLMPGPVTA	+x	YLSEGDMAA	-x
YLWPGPVTA	1	LLAGLVSLL	-1	YLVSFGVWI	+x	YMDDVVLGA	-x
FLYGALALA	1	IAATYNFAV	-1	YLYPGPVTV	+x	YMIMVKCWM	-x
FLDQVPFSV	1	YTDQVPFSV	-1	YLYVHSPAL	+x	YVITTQHWL	-x
ILWQVPFSV	1			YMLDLQPET	+x		

Table A.6: Coepra 2006 classification data of the learning and prediction set for **problem 3**

learning set				prediction set			
binding set		**nonbinding set**		**binding set**		**nonbinding set**	
molecules	class	molecules	class	molecules	class	molecules	class
AIFQCSMTK	1	ILAVERYLK	-1	ALNFPGSQK	+x	IASTPKKHR	-x
PLTFGWCYK	1	PTVRERMRR	-1	DLSHFLKEK	+x	IIATDIQTK	-x
QIYPGIKVR	1	MSKDGKKKK	-1	HLFGYSWYK	+x	ILKALGPAA	-x
AVDLSHFLK	1	TGFEAHVDK	-1	ILGLNKIVR	+x	KEEIRRIWR	-x
LLGPGRPYR	1	SITKGEKLR	-1	KIFSEVTLK	+x	KGERVDGNR	-x
GIPHPAGLK	1	RLLINKEKA	-1	KLIETYFSK	+x	KGQSASRLK	-x
GPISGHVLK	1	VIQDNSDIK	-1	KQSSKALQR	+x	KLVPVEPDK	-x
AIFQSSMTK	1	SVMEVYDGR	-1	LIYRRRLMK	+x	KMFPEVKEK	-x
RFKMFPEVK	1	SVNEPMSIY	-1	MTKILEPFR	+x	KMQVIGDQY	-x
ILRGSVAHK	1	TIGKIGNMR	-1	MVHQAISPR	+x	KNMIIKPGK	-x
MAVFIHNFK	1	TITLPCRIK	-1	NTPVFAIKK	+x	KQLTEAVQK	-x
HMYVSGKAR	1	TLTLLSVTR	-1	QIIEQLIKK	+x	KTGGPIYRR	-x
KLTEDRWNK	1	MELVRMIKR	-1	RILGTYLGR	+x	KTGKYARMR	-x
IVTDFSVIK	1	IGWPTVRER	-1	RLEDVFAGK	+x	KYLEEHPSA	-x
KIRLRPGGK	1	VQNANPDCK	-1	RLRPGGKKK	+x	LAELLGWKK	-x
TIKIGGQLK	1	TSAFVFPTK	-1	SLFRAVITK	+x	LAENREILK	-x
ATIGTAMYK	1	AAARKAACA	-1	TLFCASDAK	+x	LARSALILR	-x
ELNEALELK	1	AGASTSAGR	-1	VVGACGVGK	+x	LIRTVRLIK	-x
ATVQGQNLK	1	VMTRGRLKA	-1	YLAWVPAHK	+x	LLEYVTLKK	-x
		GGQKGRGSR	-1			LLQTGIHVR	-x
		PTVLESGTK	-1			LMHCQTTLK	-x
		INVHHYPSA	-1			LSTRGVQIA	-x
		STSAGRKRK	-1			LTQDLVQEK	-x
		LGDNQIMPK	-1			LVREIRKHK	-x
		TLSITKGEK	-1			LVRTGMDPR	-x
		KQWPLTEEK	-1			LVWMACHSA	-x
		STPRLLINK	-1			MGLLECCAR	-x
		RTEEKNFQK	-1			MGVQMQRFK	-x
		AIKKKDSTK	-1			MGYELHPDK	-x
		LMSRKHKWK	-1			MIKRGINDR	-x
		RMRRAEPAA	-1			NLNDATYQR	-x
		RTLEDNEER	-1			PAAQPKRRR	-x
		GTATLRLVK	-1			PIYRRVNGK	-x
		AADWLTSTA	-1			QARRNRRRR	-x
		PIVGAETFY	-1			QGVSIEWRK	-x
		VASGYDFER	-1			QLVFGIDVK	-x
		VSYQPLGDK	-1			QVIGDQYVK	-x
		YVTNRGRQK	-1			QVPDSDPAR	-x
		LVDFRELNK	-1			RDYVDRFYK	-x
		QTTLKYAIK	-1			RIQGKLEYR	-x
		GAITSSNTA	-1			RIVELLGRR	-x
		ITPVNSLEK	-1			RLQLSNDNR	-x
		DIQTKELQK	-1			RVTGGGAMA	-x
		ILDIRQGPK	-1			RYMQSERCR	-x
		TLPRRSGAA	-1			SAFDERRNK	-x
		LAALITPKK	-1			SEAARWNSK	-x
		ASIDAQSGA	-1			SGKARGWFY	-x
		SGHSRTTVK	-1			SIIPSGPLK	-x
		MMDQVRESR	-1			SMENTRATK	-x
		VARELHPEY	-1			SSNTAATNA	-x
		VTTERKTPR	-1			SSTLELRSR	-x
		QTLSLGSQK	-1			SSVPSYKTY	-x
		TVQGQNLKY	-1			STLPRRSGA	-x

table continued on the following page...

...table continued from previous page.

learning set				prediction set			
binding set		nonbinding set		binding set		nonbinding set	
molecules	class	molecules	class	molecules	class	molecules	class
		RLAFHHVAR	-1			AIFEISYFK	-x
		FVNTPPLVK	-1			AISPRTLNA	-x
		SWYHGPVSR	-1			AITSSNTAA	-x
		KLVSAGIRK	-1			ALAELLGWK	-x
		HQAAMQMLK	-1			ALAETSYVK	-x
		VADICKKYK	-1			ALILRGSVA	-x
		RAWCQVAQK	-1			ASTSAGRKR	-x
		RIRQRGPGR	-1			ASYLFQQDK	-x
		PIPVGEIYK	-1			ATGFKQSSK	-x
		KVAPVIKAR	-1			DIKVVPRRK	-x
		ALITPKKIK	-1			DLRVLSFIK	-x
		CFSDSAIRK	-1			EIYKRWIIL	-x
		AIRETVELR	-1			ESPSAPPHR	-x
		TLTLFNVTR	-1			FLNVIVHSA	-x
		MVLSAFDER	-1			GASTSAGRK	-x
		AVKLYRKLK	-1			GGMDFDSKK	-x
		VTTTNPLIR	-1			GGSQPPRAA	-x
		LLGHIVSPR	-1			GIKVRQLCK	-x
		NTSSSPQPK	-1			GINDRNFWR	-x
		MIMEKGEIK	-1			GLNKIVRMY	-x
		GEWFEAQTK	-1			GLVSAGIRK	-x
		ELTLEGVAR	-1			GSNLLSICK	-x
		RLQLSNGNR	-1			GSQPPRAAA	-x
		NILLQYVVK	-1			GSYFFGDNA	-x
		VALRHVVCA	-1			GTKVLPRGK	-x
		GTDSVILIK	-1			GTMVMELVR	-x
		IVHSATGFK	-1			GTRQARRNR	-x
		TILKALGPA	-1			GVFELSDEK	-x
		ASQIYPGIK	-1			GVMTRGRLK	-x
		TVSGLAWTR	-1			GYSWYKGER	-x
		KITTESIVI	-1			HIVSPRCEY	-x
		ALAALITPK	-1			HSSYLKSKK	-x
		ISPLNTSYR	-1			TLTDTTNQK	-x
		PSAGKDPKK	-1			TSAGIPDFR	-x
		AMAGASTSA	-1			TSQYRIQGK	-x
		GYVIGTQQA	-1			VARYMQSER	-x
		GAASRDLEK	-1			VGFLLLKYR	-x
		SAGRKRKSA	-1			VLSAFDERR	-x
		RYEFLWGPR	-1			VLSFIKGTK	-x

Table A.7: Coepra 2006 classification data of the learning and prediction set for **problem 4**

Bibliography

[1] Q. Zhang and Muegge I. Scaffold hopping through virtual screening using 2D and 3D similarity descriptors: ranking, voting, and consensus scoring. *Journal of Medicinal Chemistry*, 49(5):1536–1548, 2006.

[2] Hans-Georg Rammensee, Jutta Bachmann, Niels P. Emmerich, Oskar A. Bachor, and Stefan. Stevanovic. Syfpeithi: database for mhc ligands and peptide motifs. *Immunogenetics*, 50(3-4):213–219, 1999. 0093-7711 (Print) Journal Article Review.

[3] Vladimir Brusic, George Rudy, and Leonard C Harrison. Mhcpep, a database of mhc-binding peptides: update 1997. *Nucleic Acids Research*, 26:368–371, 1998.

[4] Christopher J C Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[5] Ovidiu Ivanciuc. Coepra 2006: Comparative evaluation of prediction algorithms. `http://www.coepra.org/index.html`, 2006.

[6] Kirsten Falk and Olaf. Rotzschke. The final cut: how erap1 trims mhc ligands to size. *Nat Immunol*, 3(12):1121–1122, 2002. 1529-2908 (Print) Comment News.

[7] Michelle Krogsgaard and Mark M. Davis. How T cells 'see' antigen. *Nature Immunology*, 6(3):239–245, 2005.

[8] Joan Goverman, Tim Hunkapiller, and Leroy Hood. A speculative view of the multicomponent nature of T cell antigen recognition. *Cell*, 45(4):475–484, 1986. 0092-8674 (Print) Journal Article Review.

[9] Jonathan B. Rothbard, Robert I. Lechler, Kevin Howland, Vineeta Bal, David D. Eckels, Rafick Sekaly, Eric O. Long, William R. Taylor, and Jonathan R. Lamb. Structural model of HLA-DR1 restricted T cell antigen recognition. *Cell*, 52(4):515–523, 1988. 0092-8674 (Print) Journal Article.

[10] J. Nikolich-Zugich, M. K. Slifka, and I. Messaoudi. The many important facets of T-cell repertoire diversity. *Nat Rev Immunol*, 4(2):123–132, 2004. Nikolich-Zugich, Janko Slifka, Mark K Messaoudi, Ilhem Research Support, U.S. Gov't, P.H.S. Review England Nature reviews. Immunology Nat Rev Immunol. 2004 Feb;4(2):123-32.

[11] M. Regner. Cross-reactivity in t-cell antigen recognition. *Immunol Cell Biol*, 79(2):91–100, 2001. Regner, M Review Australia Immunology and cell biology Immunol Cell Biol. 2001 Apr;79(2):91-100.

[12] Bjoern Peters, Huynh-Hoa Bui, Sune Frankild, Morten Nielsen, C. Lundegaard, E. Kostem, D. Basch, and Soren Buus. A community resource benchmarking predictions of peptide binding to MHC-I molecules. *PLOS Computational Biology*, 2(6):574–584, 2006.

[13] RA Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[14] A. Sette, S. Buus, E. Appella, J.A. Smith, R. Chesnut, C. Miles, S.M. Colon, and H.M. Grey. Prediction of major histocompatibility complex binding regions of protein antigens by sequence pattern analysis. *Proc Natl Acad Sci U S A.*, 86(9):3296–3300, 1989.

[15] Kenneth C Parker, Maria A Bednarek, and John E Coligan. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *Journal of Immunology*, 152:163–175, 1994.

[16] Irini A. Doytchinova and Darren R. Flower. Physicochemical explanation of peptide binding to hla-a*0201 major histocompatibility complex: a three-dimensional quantitative structure-activity relationship study. *Proteins: Structure, Function, and Genetics*, 48(3):505–518, 2002.

[17] Gang Wu and Edward Y Chang. Adaptive feature-space conformal transformation for imbalanced-data learning. *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, pages 816–823, 2003.

[18] M.A. Saper, P.J. Bjorkman, and D.C. Wiley. Refined structure of the human histocompatibility antigen hla-a2 at 2.6å resolution. *Journal of Molecular Biology*, 219(2):277–319, 1991.

[19] David N. Garboczi, Partho Ghosh, Ursula Utz, Qing R. Fan, William E. Biddison, and Don C. Wiley. Structure of the complex between human t-cell receptor, viral peptide and hla-a2. *Nature*, 384(6605):134–141, 1996. 0028-0836 (Print) Comment Journal Article.

[20] Zhihua Lin, Yuzhang Wu, Yunlong Wei, Bing Ni, Bo Zhu, and Li Wang. A rapid method for quantitative prediction of high affinity ctl epitopes: Qsar studies on peptides having affinity with the class i mhc molecular hla-a*0201. *Letters in Peptide Science*, 10:15–23, 2003.

[21] D. J. Brooks, J. R. Fresco, A. M. Lesk, and M. Singh. Evolution of amino acid frequencies in proteins over deep time: Inferred order of introduction of amino acids into the genetic code. *Molecular Biology and Evolution*, 19:1645–1655, 2002.

[22] Jennifer J. Kuhns, Michael A. Batalia, Shuqin Yan, and Edward J. Collins. Poor binding of a HER-2/neu epitope (GP2) to HLA-A2.1 is due to a lack of interactions with the center of the peptide. *Journal of Biological Chemistry*, 274(51):36422–36427, 1999.

[23] RCSB PDB protein data bank. `http://www.rcsb.org/pdb/home/home.do`.

[24] Jennifer Buslepp, Huanchen Wang, William E Biddison, Ettore Appella, and Edward J Collins. A correlation between tcr v docking on mhc and cd8 dependence:implications for t cell selection. *Immunity*, 19:595–606, 2003.

[25] Y.H. Ding, B.M. Baker, D.N. Garboczi, W.E. Biddison, and D.C. Wiley. Four A6-TCR/peptide/HLA-A2 structures that generate very different T cell signals are nearly identical. *Immunity*, 11(1):45–56, 1999.

[26] Y.H. Ding, K.J. Smith, D.N. Garboczi, U. Utz, W.E. Biddison, and D.C. Wiley. Two human T cell receptors bind in a similar diagonal mode to the HLA-A2/Tax peptide complex using different TCR amino acids. *Immunity*, 8(4):403–411, 1998.

[27] A.R. Khan, B.M. Baker, P. Ghosh, W.E. Biddison, and D.C. Wiley. The structure and stability of an hla-a*0201/octameric tax peptide complex with an empty conserved peptide-n-terminal binding site. *Journal of Immunology*, 164(12):6398–6405, 2000.

[28] G.F. Gao, J. Tormo, U.C. Gerth, J.R. Wyer, A.J. McMichael, D.I. Stuart, J.I. Bell, E.Y. Jones, and B.K. Jakobsen. Crystal structure of the complex between human CD8alpha(alpha) and HLA-A2. *Nature*, 387(6633):630–634, 1997.

[29] E.J. Collins, D.N. Garboczi, and D.C Wiley. Three-dimensional structure of a peptide extending from one end of a class I MHC binding site. *Nature*, 371:626–629, 1994.

[30] D.R. Madden, D.N. Garboczi, and D.C. Wiley. The antigenic identity of peptide-MHC complexes: a comparison of the conformations of five viral peptides presented by HLA-A2. *Cell*, 75(4):693–708, 1993.

[31] R.C. Hillig, P.G. Coulie, V. Stroobant, W. Saenger, A. Ziegler, and M. Hülsmeyer. High-resolution structure of HLA-A*0201 in complex with a tumour-specific antigenic peptide encoded by the MAGE-A4 gene. *Journal of Molecular Biology*, 310(5):1167–1176, 2001.

[32] GenomeNet Japan. Aaindex: Amino acid index database. `http://www.genome.jp/aaindex`.

[33] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica*, A32:922–923, 1976.

[34] Wolfgang Kabsch. A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica*, A34:827–828, 1978.

[35] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[36] T Joachims. Estimating the generalization performance of an svm efficiently. *International Conference on Machine Learning (ICML)*, 2000.

[37] T Joachims. *Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms.* Kluwer, 2002. Dissertation.

[38] T Joachims. Transductive inference for text classification using support vector machines. *International Conference on Machine Learning (ICML)*, 1999.

[39] K. Morik, P. Brockhausen, and T. Joachims. Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring. *International Conference on Machine Learning (ICML)*, 1999.

[40] B. W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim Biophys Acta*, 405(2):442–451, 1975. 0006-3002 (Print) Journal Article Research Support, U.S. Gov't, Non-P.H.S. Research Support, U.S. Gov't, P.H.S.

[41] Eispack. `http://www.netlib.org/eispack/`.

[42] Jihoon Yang and Vasant G. Honavar. Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems*, 13(2):44–49, 1998.

[43] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*, chapter 8.4, pages 174–177. MIT Press and McGraw-Hill, 2 edition, 2001.

[44] E. Corwin and A Logar. Sorting in linear time - variations on the bucket sort. *Journal of Computing Sciences in Colleges*, 20(1):197–202, 2004.

[45] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[46] Hiroshi Mamitsuka. Supervised learning of hidden markov models for sequence discrimination. *RECOMB 97*, pages 202–208, 1997.

[47] Hiroshi Mamitsuka. Predicting peptides that bind to MHC molecules using supervised learning of hidden markov models. *Proteins*, 33(4):460–474, 1998.

[48] Tin Kam Ho. Random decision forests. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, pages 278–282, Montreal, Canada, August 14-18 1995.

[49] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[50] S. Calbo, G. Guichard, P. Bousso, S. Muller, P. Kourilsky, J. P. Briand, and J. P. Abastado. Role of peptide backbone in t cell recognition. *Journal of Immunology*, 162(8):4657–4662, 1999.

[51] Markus G. Rudolph and Ian A. Wilson. The specificity of TCR/pMHC interaction. *Current Opinion in Immunology*, 14(1):52–65, 2002.

[52] David R. Musicant, Vipin Kumar, and Aysel Ozgur. Optimizing f-measure with support vector machines. *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, pages 356–360, 2003.

[53] AP Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[54] Kamalakar Gulukota, John Sidney, Alessandro Sette, and Charles DeLisi. Two complementary methods for predicting peptides binding major histocompatibility complex molecules. *Journal of Molecular Biology, JMB*, 267:1258–1267, 1997.

[55] Bjoern Peters, Sascha Bulik, Robert Tampe, Peter M. Van Endert, and Hermann-Georg Holzhutter. Identifying MHC class I epitopes by predicting the TAP transport efficiency of epitope precursors. *J Immunol*, 171(4):1741–1749, 2003. 0022-1767 (Print) Journal Article.

[56] Pierre Dönnes and Arne Elofsson. Prediction of MHC class I binding peptides, using SVMHC. *BMC Bioinformatics*, 3, 2002.

[57] Kun Yu, Nikolai Petrovsky, Christian Schönbach, Judice Y. Koh, and Vladimir Brusic. Methods for prediction of peptide binding to mhc molecules: a comparative study. *Molecular Medicine*, 8(3):137–148, 2002.

[58] Joerg Hakenberg, Alexander K. Nussbaum, Hansjoerg Schild, H. G. Rammensee, Christina Kuttler, H. G. Holzhutter, P. M. Kloetzel, Stefan H. Kaufmann, and H. J. Mollenkopf. MAPPP: MHC class I antigenic peptide processing prediction. *Applied Bioinformatics*, 2(3):155–158, 2003.

[59] Vladimir Brusic, Nikolai Petrovsky, Guanglan Zhang, and Vladimir Bajic. Prediction of promiscuous peptides that bind HLA class I molecules. *Immunology and Cell Biology*, 80(3):280–285, 2002.

[60] L. Wuju and X. Momiao. Tclass: tumor classification system based on gene expression profile. *Bioinformatics*, 18(2):325–326, 2002. 1367-4803 (Print) Journal Article Research Support, U.S. Gov't, P.H.S.

[61] Sanzo Miyazawa and Robert L. Jernigan. A new substitution matrix for protein sequence searches based on contact frequencies in protein structures. *Protein Eng.*, 6(3):267–278, 1993.

Publications

[1] H. Riedesel, B. Kolbeck, O. Schmetzer and E.W. Knapp. Peptide binding at class I major histocompatibility complex scored with linear functions and support vector machines. *Genome Inform Ser Workshop Genome Inform*, 15(1):198-212, 2004.

[2] A. Juneja, H. Riedesel, M. Hodoscek and E.W. Knapp. Bound Ligand Conformer Revealed by Flexible Structure Alignment in Absence of Crystal Structures: Indirect Drug Design Probed for HIV-1 Protease Inhibitors. *J. Chem. Theory Comput.*, 5(3):659-673, 2009

Acknowledgement

This work was accomplished in the group of Prof. E.W. Knapp at the Free University of Berlin. I want to express my gratitude to Prof. Knapp for his supervision, help and all the fruitful discussions with him. Thanks to all members of the group.

Special thanks to my friend Martin Leder and to Dr. R. Minkwitz, for proof-reading my thesis.

Furthermore I thank my grandmother and my parents for their constant support.
For the perfect cooperation with Dr. I. Muegge and Dr. F. Dicapua of Boehringer Ingelheim, Ridgefield,CT, USA I want to express my gratitude.