THE UNIVERSITY OF QUEENSLAND

School of Information Technology and Electrical Engineering

Sequence Based Prediction of Structural Disruption Caused by Crossover: a Protein Design Tool

by

Denis C. Bauer

The School of Information Technology and
Electrical engineering,
University of Queensland

Submitted for the degree of Bachelor of Science (Honours)
26th October 2005.

Denis C. Bauer
Brisbane, QLD
Tel. (040) 2813081

October 25, 2005

Head of School
School of Information Technology
and Electrical Engineering
University of Queensland
St Lucia, QLD, 4072

Dear Professor Bailes,

In accordance with the requirements of the degree of Bachelor of Science (Honours) in the School of Information Technology and Electrical Engineering, I submit the following thesis entitled

"**Sequence Based Prediction of Structural Disruption Caused by Crossover: a Protein Design Tool**".

This thesis was performed under the supervision of Dr. Mikael Bodén and Dr. Ricarda Thier.

I declare that the work submitted in this thesis is my own, except as acknowledged in the text and footnotes, and has not been previously submitted for a degree at the University of Queensland or any other institution. The results were submitted to the 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, San Diego.

Yours sincerely,

Denis C. Bauer

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor, Dr Mikael Bodén, for his support and guidance throughout the project. He has substantially improved my technical knowledge and academic writing skills.

I would like to extend my appreciation to Dr. Ricarda Thier for her valuable support and advice on the biological aspects of this work.

Special thanks to John Hawkins and Stefan Maetschke for their helpful suggestions.

I would also like to thank my parents for their love and support as well as Fabian Buske for his love, patience and confidence in me.

ABSTRACT

The project's aim is to provide a score that aids in computational protein design in producing proteins with desired function (e.g. higher effectiveness). This work examines therefore the ability and accuracy of machine learning approaches in predicting a structural disruption score from the amino acid sequence alone. The SCHEMA-score was introduced by Voigt *et al.* to quantify the structural disruption by examining each sequence position as a putative crossover site.

Unlike traditional protein design methods where new proteins are built from scratch, recombination deals with native sequence parts whose properties are already known. A pre-requisite for effective directed recombination is the knowledge of possible recombination sites.

SCHEMA divides a protein into segments that are likely to fold independently from the rest of the protein. Recombining segments from different parent proteins would leave the main structure intact while simultaneously exploring hybrid functions.

The SCHEMA calculation, however, requires the full tertiary description of the protein. This severely limits the number of proteins to the small group from which structural segments are attained. Due to the expensive and complicated nature of structure determination, the number of proteins with known structure is likely to remain small relative to the total number of known sequences.

Circumventing the need for the full structure is achieved by predicting the SCHEMA score from the primary structure. This enables the exploration of yet unresolved and even hypothetical sequences. The prediction is done by a two step approach: first predicting *secondary structure* from sequence and then predicting the SCHEMA score from the predicted *secondary structure*. Neural networks and support vector regression methods were surveyed. Bidirectional recurrent neural networks, as the best method, achieve a correlation coefficient of 0.88 for the prediction which indicates the feasibility of replacing SCHEMA with little loss of precision.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Proteins are mediators in complicated biochemical reactions within an organism. They catalyze reactions that are not only important for the organism but also of interest in industrial applications. Making use of the diverse functions of natural proteins is desirable. Recruited natural proteins, however, are often insufficient to fulfill the task as nature did not select them for being active outside the biochemical network of an organism [1]. Designing proteins appears to be necessary to meet industrial requirements.

Designing a protein implies knowledge about the mechanism of forming the protein's shape from the sequence of amino acids, because folding influences the functional properties of a protein. A new sequence, designed from scratch, must not only be stable but adopt the desired fold (reverse engineering [2]). However, design methods are not sufficiently mature to draw conclusions about the desired fold from the sequence [3]. Without guidance, the full sequence space can not be exploited effectively. Searching in a reduced sequence space seems to be more promising.

Several methods have been developed for reducing the sequence search space [4, 5, 6]. Along with the reduction, diversity often decreases as well. Finding the protein with the required properties seems to be unlikely when only choosing from similar sequences is possible. Reducing space to a systematically searchable size and offering a diversity of functional proteins at the same time would be optimal.

Besides mutating the sequence, evolution possesses another method to generate new proteins: recombination. Parts of native proteins can be exchanged to get an offspring with combined or even new properties. Recombination has proven to be very powerful in wet-lab experiments [7, 8, 9] because it offers diversity and limits the sequence space to refined native sequences whose properties are under better control than the fold of a sequence designed from scratch or modified by mutagenesis[10].

A pre-requisite for effective directed recombination is the knowledge of the borders of the blocks that can be exchanged without interfering with the structure. SCHEMA is a tool that is able to identify borders by deriving information out of the full 3D description of the protein[1]. The

assumption behind SCHEMA is that function is preserved if the main structural features are left intact. SCHEMA uses the knowledge of interactions between residues to predict the impact a recombination site would have on the global structure. The resulting borders divide the protein into blocks that are likely to fold independently without the influence of the rest of the protein. However, the full 3D description is hard to determine for some proteins. Consequently, the ability of SCHEMA is limited. Disengaging the calculation from the need of a full 3D description can overcome these limitations.

This work examines the suitability and accuracy of the predicted SCHEMA profile as the basis for future recombination sites selection. The goal was to develop a predictor that derives the SCHEMA score from the sequence of the protein only. This is a typical machine learning problem where characteristics have to be identified from a data set and applied to new unseen data. Thus, it is promising that machine learning approaches are able to solve this new problem successfully.

Two different machine learning approaches are surveyed: Neural Networks and Support Vector Regression (SVR). SCHEMA operates on the exact interactions between residues. Deriving the exact contact map from the sequence is almost as difficult as deriving the tertiary structure of a protein and is therefore likely to cause similar problems. It has been found that providing additional structural features improves the accuracy of tertiary structure prediction from sequence [11]. Besides the 3D structure, a protein has sub-structures it initially folds into. The characteristics of these sub-structures can be described by structural features [12]. Structural features such as *contact numbers*, *secondary structure* or *solvent accessibility* scores are predictable more accurately and thus add more certainty to the final prediction. There are several methods published that predicts these features from the primary structure quite accurately(e.g. *secondary structure* prediction [13, 14, 15], *contact numbers* and *solvent accessibility* prediction [16, 17]). Following these approaches, the SCHEMA score should as well be predictable from the primary structure. The prediction from predicted structural features, such as *secondary structure*, provides an alternative to the blank sequence as input for the predictor.

This work not only invests the plain sequence as input for the predictor but the combination of *secondary structure*, *contact numbers* and *solvent accessibility* scores. The additional features had to be predicted themselves because the goal is to present the sequence only as input to the predictor complex. This work presents results (e.g. a correlation coefficient of 0.88 for the best method) which suggest the ability of the developed predictor.

2. BACKGROUND MATERIAL

2.1 Computational Protein Design

Proteins have to fulfill a wide range of tasks in the organism: building the scaffold of a cell, functioning as messenger in inter cell communication or aiding in immune system responses by tagging cells. Since all these tasks are based on the interaction in a 3D space, the global shape of the protein plays a crucial role for its function. This global shape for a single protein is the tertiary structure (see Figure 2.1). A protein sequence, also known as the primary structure, folds first into the *secondary structure* which then in turn folds into the tertiary structure.

Protein design from scratch is ultimately about constructing an amino acid sequence which folds in a tertiary structure with desired properties. The number of resulting protein sequences is extremely high because of the combinatorial possibilities: for a small protein with a length of 10 residues there would exist 20^{10} sequences. Obviously it is not possible to construct each protein biochemically, to verify the functionality. Though it is necessary to determine the function in wet-lab experiments, it is possible to preselect proteins by the fold only. The fold can be approximated by computational methods, thus designing proteins *in-silico* is possible.

2.1.1 Designing Proteins from Scratch

The design of the sequence is driven by the desire to accomplish a certain function and therefore a certain fold. This approach is called reverse engineering and demands two tasks that are interdependent: Exploring sequence space to design a new protein and knowing how this sequence eventually folds. Drawing conclusions about the fold is a challenge that has not been mastered sufficiently yet. Many methods were developed ranging from simulating each possible conformation, driven by the knowledge that a protein adopts the conformation with the lowest energy(BlueGene, Folding@home), to machine learning approaches(Homology modeling [18], Lego prediction [19], and *de novo* prediction [20][21][22][23][24])(see Figure 2.2). While the simulation approaches are computationally costly, machine learning attempts are too inaccurate.

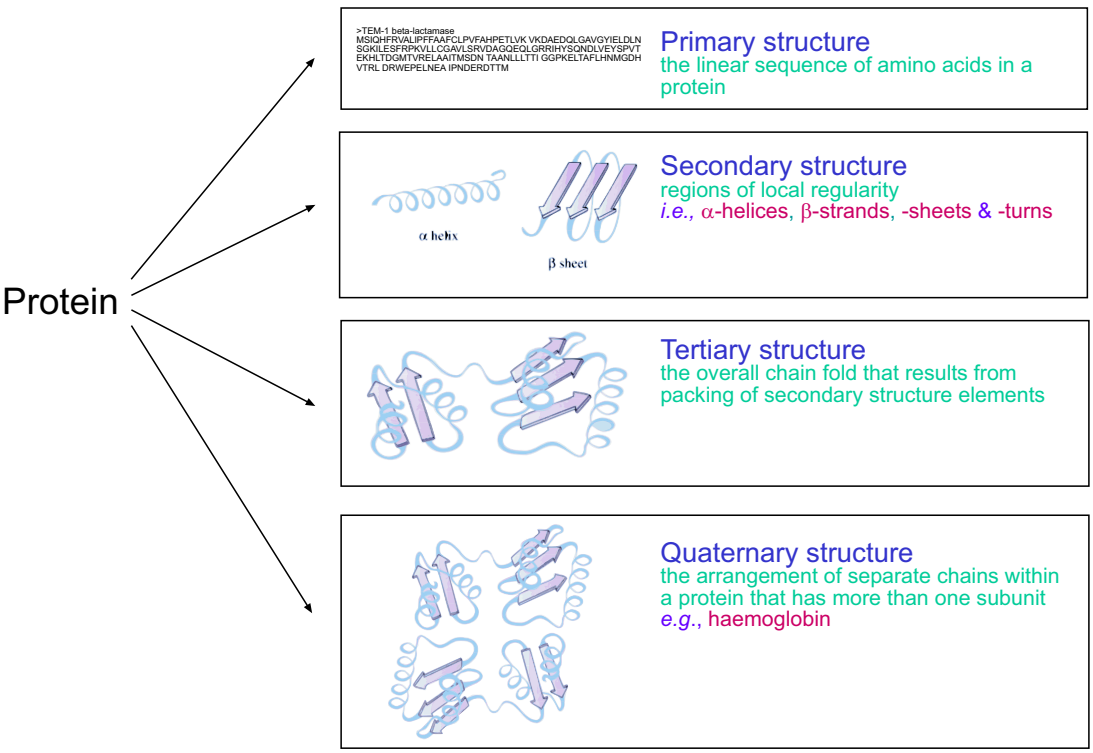


Fig. 2.1: Different folding levels of a protein. Pictures and definitions from EMBO Bioinformatics Workshop, Ireland

Attempts have been made to try to improve the latter by predicting additional structural features prior to the tertiary structure prediction. Structural features are characteristics of the tertiary structure such as *secondary structure*. Predicting structural features prior to the tertiary structure prediction is more accurate, because structural features are easier to predict and provide therefore a higher amount of certainty to the final prediction (e.g. SCRATCH[11]) [12, 25].

In many cases, functionality and nonfunctionality are separated by differences of only fractions of Ångstroms in the position of certain key atoms; an accuracy threshold well beyond the current modeling state-of-the-art. This arises the question if it would be better to reduce the search space to promising candidate sequences. The fact that not every possible amino acid sequence exists [19] makes it even more doubtful if it is necessary to have the full sequence space at one's disposal.

2.2 *Reducing Sequence Space*

Sequence space is reduced by a broad range of natural restrictions such as hydrophobic patterning [26]; however knowledge of restrictions is too limited at this stage to direct the development. Instead a standard approach is sampling the sequence space for a given fixed protein-backbone and a library

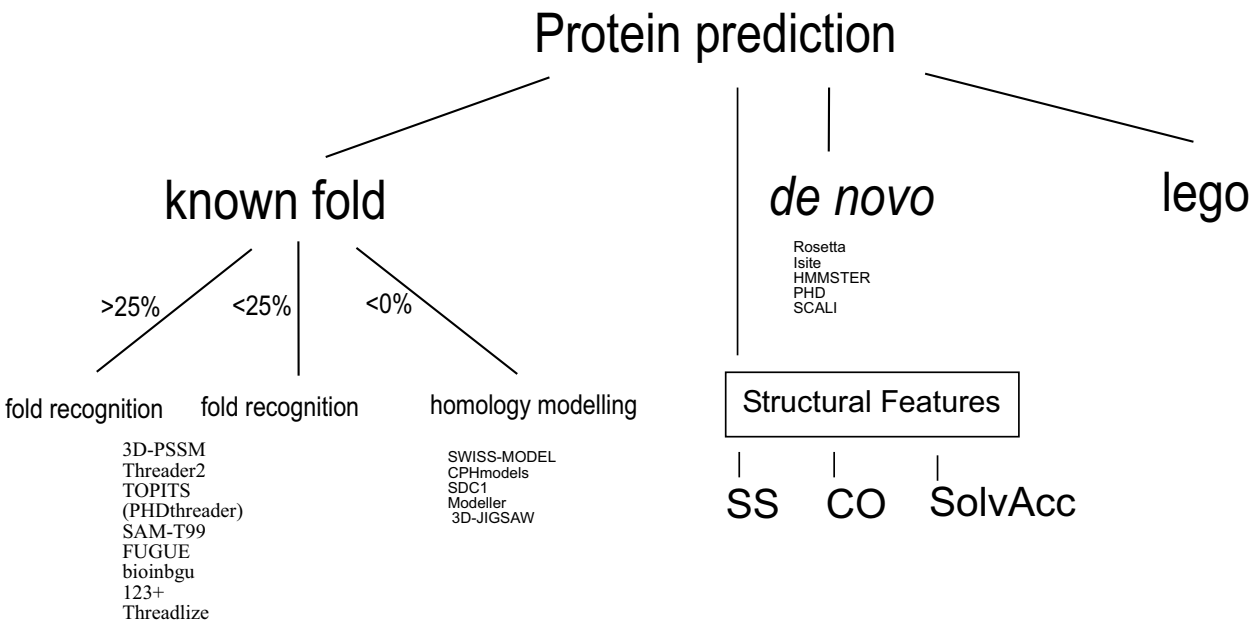


Fig. 2.2: Tertiary structure predictors can be divided into four classes. The first class applies the structure of proteins for which the fold is already known to new sequences. This branch can be sub divided according to the similarity of the sequences they are working with. If similarity in the sequence is given the fold can be directly applied from the similar sequence for which the fold is known. Homology modeling comes into the picture when there is little or non similarity of the new sequence to any sequences which have a known fold. The next class is lego prediction where proteins for which the fold is known are cut into small sequence parts out of which any new sequence is re-assembled. Along with the sequence reassembly the fold from the sequences is composed as well. Final adjustments improve the composed results. The third class is *de novo* prediction which tries with Hidden-Markov-Models or machine learning approaches to construct the structure from the sequence alone. Those methods can be supported by initially predicting structural features such as *secondary structure* (SS), *contact numbers* (CO) or *solvent accessibility* score (SolvAcc).

of different rotamers (conformations generated by discrete rotations around the side-chain torsion angles) for each amino acid. Therefore several optimization techniques have to be employed, like stochastic methods such as Monte-Carlo simulated annealing [27, 28] and various others [29, 30]. The first successful computational design of a full protein was achieved with a deterministic branch-and-bound technique based on the dead-end elimination theorem [2, 31]. Even in the case of dead-end elimination, however, heuristics must be incorporated to make convergence reasonably fast for large proteins.

2.2.1 Directed Evolution Design

A different approach to protein engineering methods is to reduce sequence space by refining natural protein sequences, such as the directed evolution approach. Directed evolution methods mimic the process of Darwinian evolution by requiring only a fraction of time [32]. One branch of directed evolution is mutation where a small number of changes is undertaken randomly to explore the neighborhood of the original sequence. The other branch is recombination: sequence-parts from different parents are merged to receive proteins with combined or new features. The idea behind recombination approaches is to divide two or more parental proteins into sequence parts which can then be rearranged to form a new hybrid protein. The hybrid may hold improved functionality due to the combined properties of its parent proteins. The sequence space is effectively reduced to the parental sequences and the possibilities of combining them. It is believed that complex proteins are formed out of smaller domains developed earlier in evolution. Those blocks or schema are pieced together by gene-duplication and recombination [33, 34, 35, 36]. In wet-lab experiments recombination has proven to be more effective than mutations [37, 38].

Recombinatorial Libraries

The best nexus can be found by producing large libraries containing every possible crossover combination. However, this is not feasible because of the large number of combinatorial possibilities. The goal is to develop a tool that does not need a library of all possibilities to fine the best combination. Several methods have been introduced that are able to create restricted evolutionary libraries [4, 5, 6]. These methods, however, generate large numbers of non-functional sequences, because they not only use recombination but undirected mutations, insertions and deletions [8]. Voigt *et al.* proposed a tool, SCHEMA [1], that divides the protein into fragments on the basis of information drawn from the tertiary structure. The sequence within one fragment has to be inherited from the

same parent in order to preserve functionally important structure parts. Not only is the library size reduced to mainly functional proteins, but it is also possible to control the location or frequencies of crossover sites, which minimizes structural disruptions and simultaneously maximize the resulting sequence diversity.

Functionality of SCHEMA

New proteins can be derived by merging sequence parts from two or more proteins. For the hybrid to inherit structural features the parental sequences must not differ too much from each other, otherwise the overall stability is risked [1]. The idea is to pass on functional blocks without disruption. The structure of a protein can be divided in global and local structure. While the global structure has mainly the function to provide a stable scaffold, the local structure forms domains that fulfill protein specific tasks.

The number of residue interactions in the parental proteins are calculated and compared to those conserved in the new protein. The number of interruptions E is calculated for each hybrid.

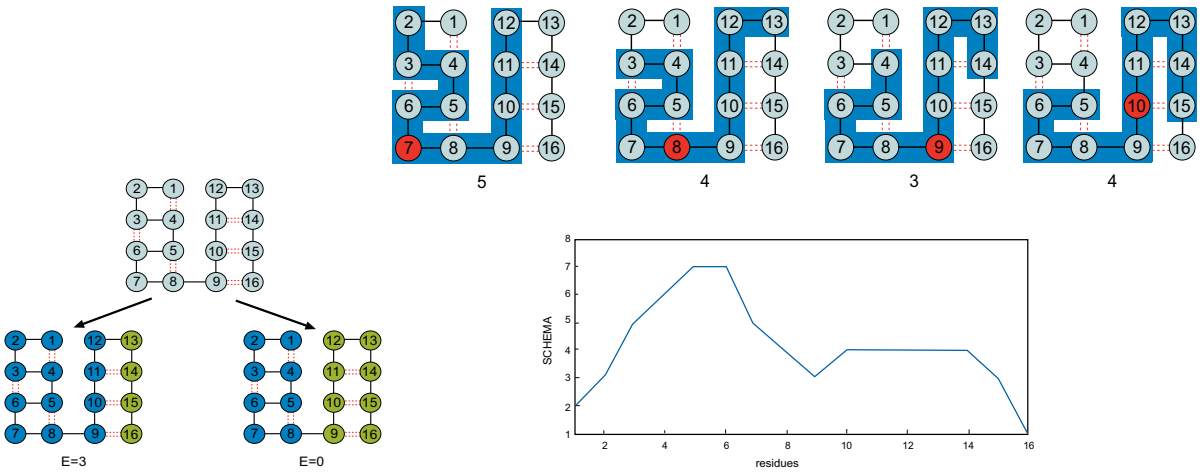
$$E_{\alpha\beta} = \sum_{i\in\alpha}\sum_{j\in\beta} c_{ij}P_{ij} \tag{2.1}$$

Where α is/are the element/s inherited from parent A and β is/are the element/s inherited from parent B. $c_{ij} = 1$ if residues i and j are within a given distance (here 4.5$\mathring{A}$) and $c_{ij} = 0$ otherwise. P is the probability that indicates the disruption. No disruption if the amino acid of the residue pair i,j in the set of potential hybrids are the same as in any of the parents, given that the i and j are in contact in the hybrid.

The assumption behind SCHEMA is that function is preserved if local structure is preserved because the global structure is provided by the similarity of both parental sequences [9]. With this assumption it is sufficient to limit the calculation of broken structure elements within the hybrid to a fixed window covering the local interaction range (see Figure 2.3). In order to retrieve a continuous measure for each residue, a SCHEMA profile S is generated that increments S for each residue i within the window w by the number of disruptions created by a certain crossover in the region.

$$S_i = \sum_{j=i-w+1}^{i}\sum_{k=j}^{j+w-2}\sum_{l=k+1}^{j+w-1} c_{kl}P_{kl} \tag{2.2}$$

$c_{kl} = 1$ if residues k and l are within a given distance and $c_{ij} = 0$ otherwise.



(a) (b)

Fig. 2.3: Window approach in SCHEMA. Circles represent amino acid residues, solid lines are the backbone contacts of directly adjacent residues, dotted lines represent the interactions of residues with amino acid beyond the direct neighborhood. (a) Demonstration of different recombination sites. UP initial structure DOWN LEFT recombination site situated at the position with the amino acid with the lessfewest *contact numbers*. 3 interactions are broken. DOWN RIGHT cut at the position SCHEMA suggested. (b) Visualization of the calculations within SCHEMA. UP red highlighted amino acids represent the position for which the SCHEMA score is calculated in the current iteration. The shaded amino acids are examined for the current calculation. Underneath each iteration picture the calculated SCHEMA score is shown. DOWN resulting SCHEMA profile for each position, the suggested position for the recombination site here is therefore 9. The window size used was 6.

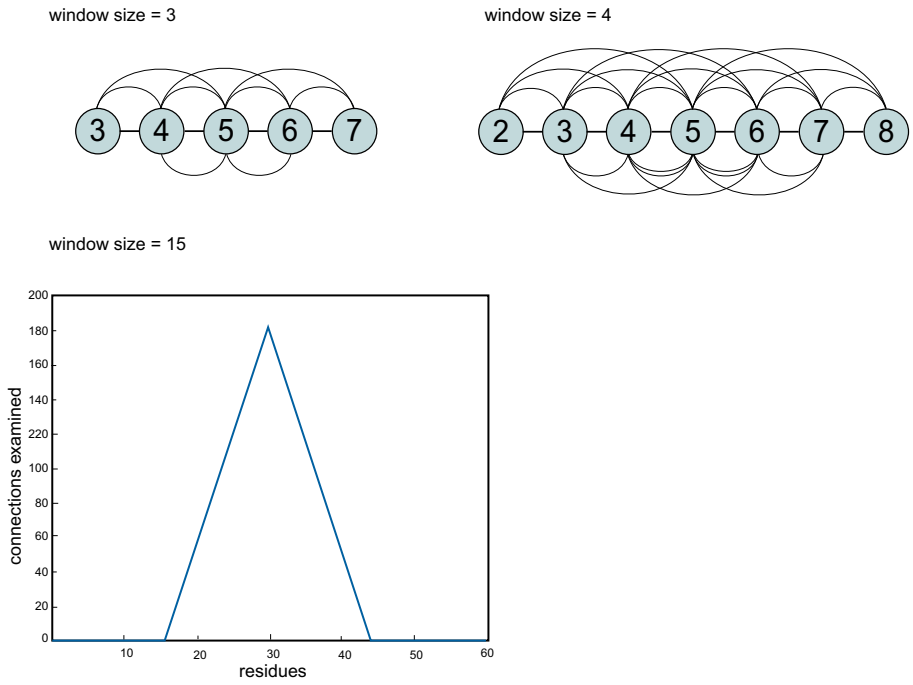


Fig. 2.4: Demonstration of the implicit weighting of interaction depended on the distance to the calculation site. UP all interaction shown below the residues are the additional ones that form the emphasis on the inner interactions. DOWN The diagram shows for each position the number of times the residue was examined

The way the sums are nested creates an implicit weighting (see Figure 2.4). Contacts, where both residues are close to i, are weighted the highest and the influence decreases if one or both contact partners are located close to the boundary of the window. $c_{kl} = 1$ if residues k and l are in contact and $c_{kl} = 0$ otherwise. Minima in the SCHEMA profile suggest the cutting sites which preserve the maximum number of internal interactions.

SCHEMA's need of the tertiary structure for the calculation is a major disadvantage. The tertiary structure is not derived for every protein. In fact relating the number of the known protein 178022 (swiss prot [39]) with the known tertiary structures 30263 (pdb [40]) shows that approximately the structure of only every 5th protein is known. The reason for this gap is the complex crystallization process: only from a reasonably large protein (> 0.1 mm in its smallest dimension), can a crystal be grown. An X-ray beam will be diffracted into a pattern of reflections. From this pattern a three dimensional map of the molecule's electron density can be produced through crystallographic data analysis. Only a very good protein crystal will produce a usable electron density pattern. Achieving such a quality is difficult especially for membrane proteins which form $20 - 30\%$ of all protein fractions in a typical genome. NMR spectroscopy does not require a crystal structure. Bringing protein in a sufficient concentration renders NMR ineffective on some proteins as well.

Both strategies suffer from being time consuming and expensive. The number of proteins with known structure is therefore likely to remain small relative to the total number of known sequences.

Being able to freely choose candidate proteins on the basis of functional properties (say, specific enzymatic activity), and not be limited to those for which full structural information is available, is highly desirable. Not until recombination site derivation is disengaged from the tertiary structure, can we fully tap the power of *in silico* protein design.

The need of the tertiary structure can be circumvent by predicting the SCHEMA score from the primary structure by using machine learning approaches.

2.3 Machine Learning Approaches for Structure Prediction

2.3.1 Approaches

A machine learns whenever it changes its structure or program in such a manner that its expected future performance improves. Observing specific characteristics of a set of data in order to recognize familiar features in a new data set is the key idea in machine learning. Two major approaches were previously reported to be successful in machine learning task involving protein sequence: Neural networks and Support Vector Regression (SVR).

Neural Networks

Inspired by the neurons in the brain, Rosenblatt developed the concept of neural networks, where a network of single calculation units (perceptron) generate a prediction for each data point which is then refined by changing the parameters of this network [41].

A perceptron has input- and output connections with one additional input enabling the introduction of a bias. Each input connection has a weight assigned which is updated in the process of learning. Each output connection possesses an activation function that is either the identity (linear output function) or transforms the node's output, e.g. to a 0 1 bounded interval (tanh) . Figure 2.5 shows the simplest version with only one perceptron in the network. The number of input connections depends on the number of input features for a single data point. When the network is trained, input is fed through the input connections and forms in the neuron a weighted sum.

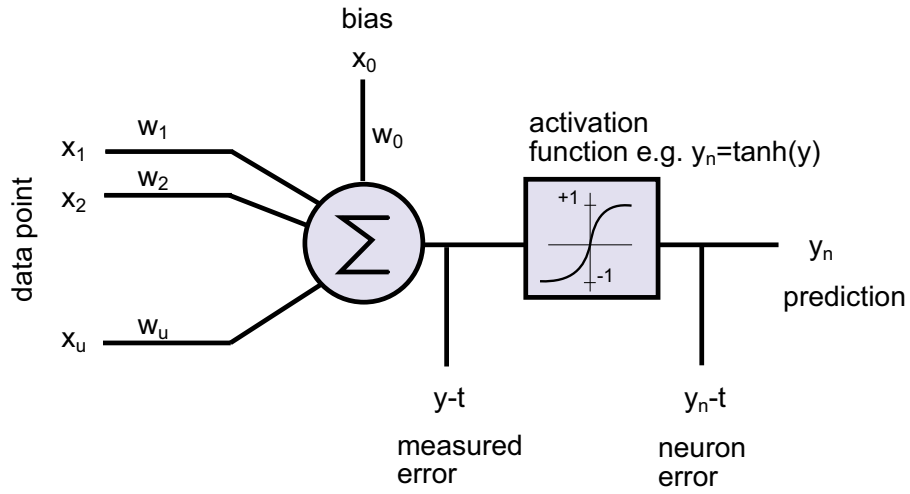


Fig. 2.5: Simplest neural network with one node and output function. The two different ways to measure the calculation error are shown, where measured error is the neuron's output before the transformation of the activation function (delta rule) and neuron error is the transformed value identical to the final output.

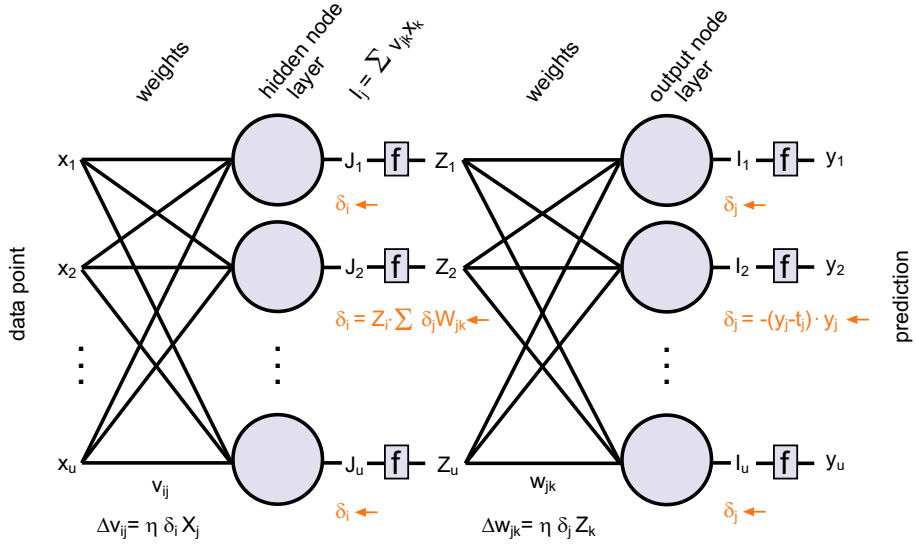


Fig. 2.6: A neural network with output layer and hidden layer. Values in orange are the error that is back propagated through the network in the weight update process.

$$y = \sum_{i=0}^{u} w_i \cdot x_i \tag{2.3}$$

After this initial prediction the actual learning phase follows where the weights are updated according to the error the prediction causes compared with the desired result.

$$E = \frac{1}{2} \sum_{p} (t_p - y_p)^2 \tag{2.4}$$

where p is the number of units.

The training process is an optimization task where the goal is to find the optimal weight vector so that the error the network produces for the input data is minimal. The weights therefore are updated in the opposite direction of the gradient of the error surface (Gradient Descent).

The perceptron update rule includes the final output of the perceptron (after the activation function see Figure 2.5 ϵ_n) in the update formula, whereas the delta rule includes the actual output of the neuron. Having access to the actual weighted sum has the benefit that the gradient descent process will be stopped not only when it has found the minimum but also, if the problem is not exactly solvable, when it is the closest to the minimum.

$$\Delta w = 2\eta(t - y) \cdot x \tag{2.5}$$

where η is the learning rate which determines how large the update steps are.

Functions that can not be approximated by a straight line can not be solved by the simple one layer approach discussed so far. An additional layer, called hidden layer, provides an additional degree of freedom (see Figure 2.6). The output of every unit in the first layer is fed in as input to every unit in the second layer.

The hidden layer, however, requires a chance in the update rule of the model because only the final target output is known, but for a weight update in the first layer a target is required as well. The Back-Propagation-Algorithm solves this problem by propagating the final output back through the network. First the output of each layer is calculated by moving through the network (Feed-Forward).

Then the error is propagated back beginning from the known target (Back-Propagation). With the propagated error, the weight vector of every layer is updated.

For neuronal network approaches the search for optimal parameters is mainly empirical. Besides

Activation hidden layer

$$J_i = \sum v_{ij} X_j \qquad I_i = f(J_i)$$

Activation output layer

$$I_j = \sum w_{jk} Z_j \qquad y_j = f(I_j)$$

Update weights going to output layer units

$$\Delta w_{jk} = \eta \delta_j Z_k \qquad \delta_j = -(y_1 - t_i) \cdot y_i$$

Update weights going to hidden output layer

$$\Delta v_{ij} = \eta \delta_i X_j \qquad \delta_i = Z_i \sum_k \delta_k w_{ki}$$

the fixed number of input nodes (dependent on the number of input features) and number of output nodes (dependent on number of output features; classification of two classes = 1) all parameters are free to choose and depends on the problem. The quality of the results strongly depend on the right choice.

Support vector Regression

Support Vector machines are becoming more and more popular for various machine learning fields. Besides the task of finding a suitable Kernel function, where the number of significantly different ones is limited, there is no additional parameter space to explore. This forms the major advantage in contrast to neuronal networks where the number of hidden units, the choice of initial weights and the learning function influences the results.

The goal in Regression is to find a linear hypothesis function $f(x) = w \cdot x + b$ that describes the given data set best. A tube, which is wrapped around the hypothesis function, of fixed size $f(x) \pm \epsilon$ should include all data points. This approach is called ϵ-Support Vector Regression (ϵ-SVR) [42].

In SVR the task is to minimize the tube size for a given data set. For each data point an error is calculated according to the loss function and the distance from the tube border. In order to minimize the tube size, it is adjuvant to allow a soft transition in the error function between data points inside the tube – no error – and the data points outside – training error. This approach is called soft margin and holds the benefit of minimizing the margin most without causing a training error that is too high (Figure 2.7).

The penalty value for each data point is given by slack variables ξ_i, ξ_i^* that are zero for data points inside the tube and increase according to a loss function and the distance to the margin for data

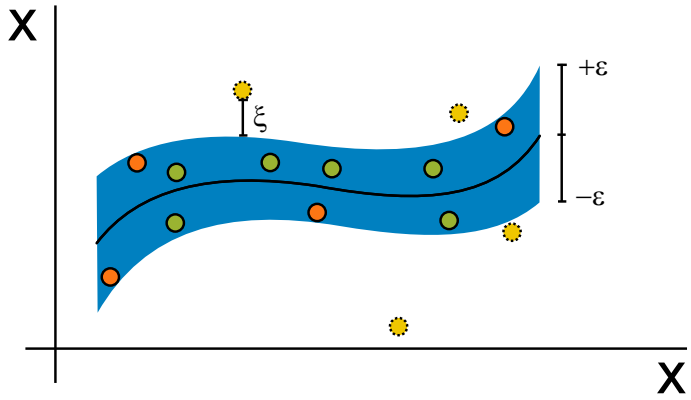


Fig. 2.7: The task in SVR is to minimize the tube around the hypothesis function. This can cause problems because it leads to very high training errors unless the data points near the tube border cause not the same error value than data points that are far off. The value is set for each data point according to the loss function and the distance from the tube border. Green are the data points not causing an error, yellow the ones causing an error and orange are the support vectors.

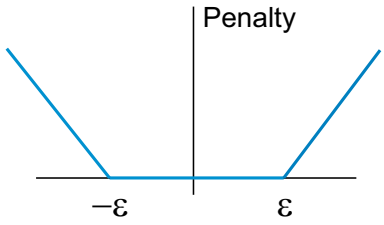


Fig. 2.8: The loss function returns 0 for data points within the tube (ϵ and -ϵ) and increase with the distance from the tube border for all other data points. The ϵ-insensitive function is to be preferred over the quadratic loss function because it is less sensitive to outliers

points outside (Figure 2.8).

Especially for protein feature prediction it is beneficial to map the data points into a higher dimensional space, called feature space, because the additional degree of freedom is all that makes it possible to approximate the data points by a straight line (Figure 2.9).

Essentially, support vector regression operates by finding the so-called support vectors that collectively represent the function in a feature space by defining the borders of the tube so that both ϵ and the error caused by data points outside the tube are minimal.

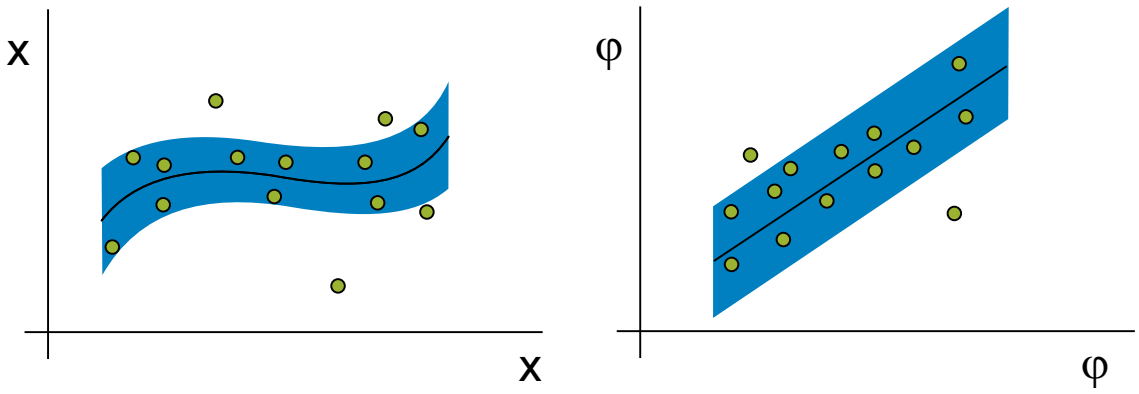


Fig. 2.9: Mapping in the feature space to transform the function in a straight line to simplify the approximation

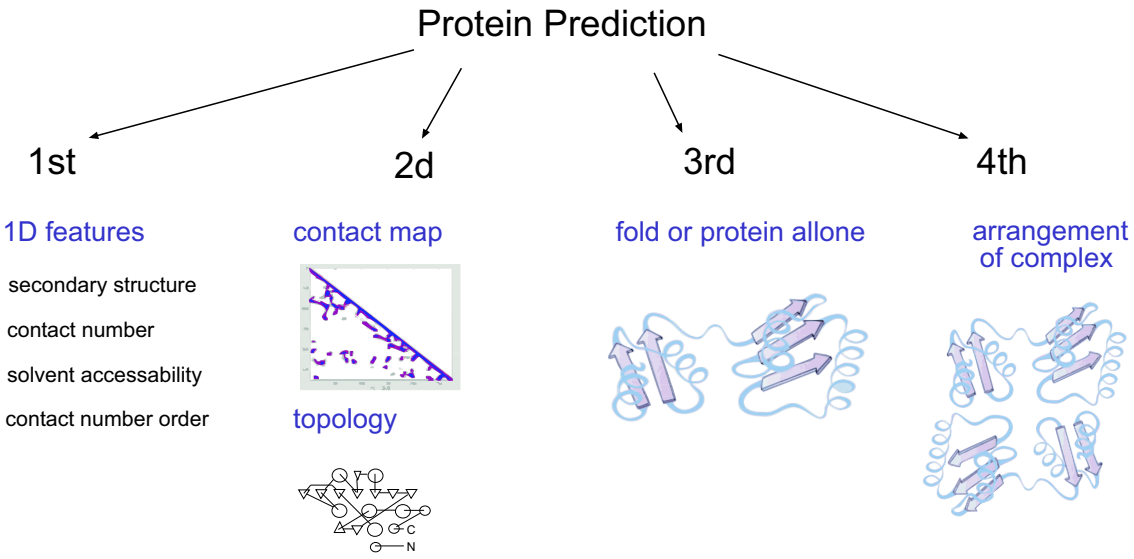


Fig. 2.10: Different structural elements that are important for prediction

2.3.2 *Examples of Structure Prediction*

In the following section the general prediction process underlaying the SCHEMA score prediction is exemplified on predictors operating in the three areas involving a structural feature: *secondary structure* prediction, *solvent accessibility* prediction and *contact number* prediction (see Figures 2.10). Predicted structural features aid in tertiary structure prediction, as mentioned earlier. However, the predictors in the following section are discussed not only to introduce general idea of the prediction process, but also because one predictor in each section was used later on to generate additional information to improve the prediction accuracy in the SCHEMA score prediction.

Secondary Structure

Amino acid sequences fold into low level structures such as α-helix, β-sheet (or strand) and coil, which is a label for other folding patterns. A more detailed classification can be made by grouping them into eight states [43]. These low-level structures eventually form the tertiary structure. Reaching an accurate prediction of around 80% has taken more than 40 years development. First and second generation predictors were based on single amino acid propensities and propensities of 3-15 adjacent amino acids [44]. The first breakthrough with accuracy above 70% was achieved by the third generation predictors by using the information contained in multiple alignments [44]. Originally the input patterns for machine learning methods were coded as vectors of concatenated numbers uniquely selected for each amino acid. Jones revolutionised input coding by using PSI-BLAST profiles[45]. The benefit of profiles is that one position can hold more than one letter, namely all amino acids that cause the same fold coded as probabilities. PSI-BLAST profiles, however, require a careful selection of the proteins participating in the multiple alignment, because an unrelated sequence would pollute the profile and lead to wrong probabilities at each position. Restricting to related proteins increases the accuracy of the predictor [19]. Further improvement can be achieved mainly by extending the database volume or through selective database search, because the predictors' accuracy strongly depends on the quality of input data presented [19].

Just as dropping quality can be caused by choosing the wrong data, too optimistic accuracy rates can generated by correlated sets [19]. Because of nonuniform testing protocols and the quite high number of predictors published lately, Rost *et al.* [46] established an automatic evaluation server (EVA) that sends a random and unbiased dataset (the n newest experimental structure added to PDB) to all applied predictors and ranks the result. See table 2.1.

Previously increasing the accuracy was expected to come along with the increment of structure databases. Recent findings suggest that PDB contains enough structural information for machine learning methods to learn sufficiently and further incensement are unlikely to introduce significantly new folding characteristics [47]. A frequently successful method of improving the accuracy is processing the result from different predicting models. However, often the best method is better than averaging over many [19].

Currently, the best methods are based on machine learning approaches, in particular neuronal network architectures (first four methods in table 2.1). The general approach is to move a window over the sequence and predicting the structure the centred amino acid participates in. The window size has to be small to avoid pollution and overfitting. However, the smaller the window size the less

predictor	average performance (Q3)	developer
SSpro1	79.1	Baldi
SSpro4	78.3	Rost and Baldi
SABLE2	76.8	Meller
PSIpred	76.2	Jones
SAM-T99sec	76.0	Karplus

Tab. 2.1: EVA: ranking of currently five best *secondary structure* predictors. Accuracy given by Q3 measure. The continuous *secondary structure* prediction used for the SCHEMA score predictor was $Q3 = 77.8$ [15].

Qindex: (Qhelix, Qstrand, Qcoil, Q3) gives percentage of residues predicted correctly as helix, strand, coil or for all three conformational states. Data downloaded in first week of March 2005.

amino acids up and downstream can influence the prediction. Interactions with residue, situated distantly from the prediction site are highly important, especially in β-sheets [48]. Baldi *et al.*[3] came up with a solution for taking long term dependencies into account: a recurrent neuronal networks (RNN) can work with information that is situated far off downstrand because it has the ability to capture distant information in the form of contextual knowledge[49, 2]. However, RNNs are not sufficient as information situated upstrand is not taken into account. For SSpro, Baldi *et al.* developed the model of a bidirectional recurrent network (BRNN) which heads the current rankings with an accuracy of 79.1% [3].

A different approach to assign each residue to one of the 3 or 8 stages was undertaken by Bodén *et al.* by introducing a Continuum Secondary Structure Predictor [15]. Following the scheme proposed by Andersen *et al.* [50] the continuous *secondary structure* predictor returns for each residue its probability to be in each state. This reflects the real behavior of the protein more accurately. While some sections of the protein are stabile after the protein is folded others are flexible through out their existence. The classification accuracy achieved by thresholding is $Q_3 = 77.3$.

Solvent Accessibility

Each residue in a protein can be predicted as buried or exposed. Whether a residue is buried or not indicates the residues' impact on the stability and the function of the protein. High conformational fluctuation is a problem among designed proteins and is influenced by solvent. The solvent properties of amino acid depend on multiple factors including van der Waals forces, hydrogen bonding, electrostatic interactions and solvent degrees of freedom. In a biological environment globular

Method	Threshold			
	0%	5%	9%	16%
Rost and Sander (PHDacc)[60]	86.0	N/A	74.6	75.0
Gianese *et al.* (PP)[58]	N/A	N/A	76.8	75.1
Kim and Park (single-stage SVM) [61]	86.2	79.8	N/A	77.8
Nguyen *et al.* Two-stage SVMs [51]	90.2	83.5	81.3	79.4

Tab. 2.2: Comparison of Performances based on a specific data set; Tabelle taken from Nguyen *et al.*[51]. N/A indicates that the corresponding result was not available from the literature. /newline The relative solvent accessibility value that determines whether a given amino acid is buried or exposed can be adjusted and has an influence on the prediction accuracy. An amino acid is declared not exposed to solvent when its observed accessibility is less than a fraction of its accessibility observed in a reference state. Usual values are around 20%. The range available here goes from 0% (totally buried) to 55% (very exposed).

proteins bury hydrophobic residues in their interiors and expose hydrophilic residues to solvent. It is believed that buried polar residues destabilize the protein. This concept is so strong that constructing sequences with hydrophobic amino acid at defined positions (hydrophobic patterning) is sufficient to design new stable proteins [26].

The different approaches of predicting solvent accessibility can be divided into the following categories [51]: Bayesian, neural networks, information theoretical approaches and support-vector-machine (SVM).

Bayesian methods take local interactions among amino acid residues into account by extracting the information from single or multiple sequence alignments in order to obtain posterior probabilities for the final prediction [52]. The input for neural networks is residues in a local neighbourhood. By finding an arbitrary, nonlinear mapping they can predict the state of a residue at a particular location [53, 54, 55, 56]. Mutual information between sequence and solvent accessibility values derived from a single amino acid residue, or pairs of residues in the neighbourhood, is used by information theoretical approaches [57]. Recent variants of these approaches use probability profiles derived from amino acid residues in the neighbourhood of the predicting site [58] or search for continuous approximations for the solvent accessibility value [59].

Nguyen *et al.* claim to reach with a two-stage SVMs an accuracy of over 90.2%, which would make their method the best published so far [51]; see table 2.2.

Contact Number

In the tertiary structure each residue has contact with a specific number of residues not only in the immediate sequential neighbourhood but sometimes far off up- or downstream. Contact means in this case being in the spatial surrounding within a specific distance [62]. Given a sequence of *contact numbers* along the chain, there is only a limited number of conformations that can fulfill the constraint [63]. Despite this close relation, *contact numbers* are not predicted sufficiently well yet.

The first method was a heuristic method that predicts a continuous value for *contact numbers*. Recent approaches are more successful with a 2 state prediction. Pollastri *et al.* had been successful with BRNNs here as well, and reaches an accuracy of 73.9% [55]. A SVR approach for *contact number* prediction was introduced by Yuan and reaches a correlation coefficient of 0.70 [17].

2.4 Summary of the Background Material

SCHEMA appears to be a very successful tool for protein design attempts [9]. It divides protein sequence in sections that can fold independently from the rest of the structure. SCHEMA provides therefore the means to recombine sequence parts of different proteins by keeping the main structural features intact. This promises to preserve the initial function but holds the possibility to increase performance because the offspring combines features of the parental proteins [1].

SCHEMA needs the tertiary structure to calculate the recombination site positions which limits the protein design attempts to those proteins for which the tertiary structure is already known. The limitation can be overcome by predicting SCHEMA from the sequence.

Machine learning methods, such as neural networks and SVR, have previously found to be useful for the prediction of structural features from sequence. Since SCHEMA takes the interactions of residues into account which is a sequence feature, predicting the SCHEMA score with similar approaches should be possible.

3. MATERIALS AND METHODS

3.1 Data set

A data set consisting of 945 proteins taken from the Protein Data Bank was used. The data set represents a diverse range of proteins and has no pairs with more than 25% sequence similarity [64]. This data set is later referred to as the primary data set.

Additionally, a second data set was used to verify the results. The Nh3d data set 70% of folds at the topology level of the CATH database and represents more than 90% of the structures in the PDB that have been classified by CATH. It seems to be even more free of sequence similarity than the first data-set [65]. It will be referred to as Nh3d.

A third data set was used to train the *solvent accessibility* predictor on. It was developed by Carugo for predicting the binary state for each each residue to be buried or exposed [66].

3.2 The target SCHEMA score

The SCHEMA score was determined for the proteins in the data set using the equation given by Arnold *et al.* [1].

$$S_i = \sum_{j=i-w+1}^{i} \sum_{k=j}^{j+w-2} \sum_{l=k+1}^{j+w-1} c_{kl} P_{kl} \tag{3.1}$$

S_i describes, for each residue, the number of contacts within the window $(i-w, i+w)$ that would be broken if the recombination site is positioned at i.

For recombination site determination with two or more parents, aiming to produce hybrid structures, the sequence composition of all parents should be taken into consideration. Hence, $P = 0$ if amino acids at k and l are identical in all parents. In the tests presented herein P has always the value of 1, since the SCHEMA score is calculated for a single protein.

The window size is set to 14 to follow the configuration reported by Voigt *et al.*

>1a3aa Atomes included CA—CB— C
L 1
F 1 1
K 0 1 1
L 0 0 1 1
G 0 0 0 1 1
A 0 0 0 0 1 1
E 0 0 0 0 1 1 1
N 0 0 0 0 0 1 1 1
I 0 0 0 0 0 1 0 1 1
F 0 0 0 0 0 0 0 0 1 1
L 0 0 0 0 0 0 0 0 0 1 1
G 0 0 0 0 0 0 0 0 0 0 1 1
R 0 0 0 0 0 0 0 0 0 0 0 1 1
K 0 0 0 0 0 0 0 0 0 0 0 0 1 1
A 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1
A 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1

Tab. 3.1: Map from 1A3A, primary data set. Contact Maps are symmetric therefore only the lower half was stored. 1 indicates that the amino acid from the horizontal position is with the one vertical in contact.

However, the configuration used to derive the contact map was not specified by Voigt *et al*[1]. The contact map holds the information about which amino acids are in contact and forming the tertiary structure of the protein. Thus, several experiments were necessary to be as close to the SCHEMA score (used by Voigt *et al.*) as possible. The output of a online program was used as reference profile. The program was accessible over the webpage from the research group of Frances Arnold[1].

3.2.1 Contact Map

The complete binary contact map was derived for each of the proteins in the data set from the pdb file. A contact map is a binary table that holds the information which amino acids are in contact (see Table 3.1). Contact is defined by being in the Euclidian cut-off distance of 4.5 Å. A PDB file holds the coordinates for each atom the amino acid is built of (see Table 3.2). Several possibilities arise to determine which atoms and therefore which amino acids are in contact.

An examination of the most likely combination reveled that Arnold *et al.* used the most common configuration, namely including the C, C-α atom and if existing the C-β atom (see Figure 3.1(a)

[1] http://www.che.caltech.edu/groups/fha/ available until approx. Sept 2005

Record name	A #	A Name	AA Name	Chain		x	y	z			
ATOM	1	N	LEU	A	4	-3.883	41.780	40.071	1.00	37.29	N
ATOM	2	CA	LEU	A	4	-4.394	42.817	39.132	1.00	36.87	C
ATOM	3	C	LEU	A	4	-5.413	42.211	38.178	1.00	35.97	C
ATOM	4	O	LEU	A	4	-5.799	42.876	37.208	1.00	36.91	O
ATOM	5	CB	LEU	A	4	-3.265	43.498	38.355	1.00	38.51	C
ATOM	6	CG	LEU	A	4	-3.604	44.793	37.605	1.00	39.28	C
ATOM	7	CD1	LEU	A	4	-4.290	45.783	38.535	1.00	38.14	C
ATOM	8	CD2	LEU	A	4	-2.372	45.423	36.968	1.00	38.75	C

Tab. 3.2: Extract from a PDP file. First amino acid from 1A3A, primary data set. *A* is the abbreviation for atom and *AA* for amino acid. X,y,z are the coordinates. The other columns displayed here were not used; definitions and descriptions can be found in "PDB Format Description Version 2.2" on www.rcsb.org/pdb.

ϵ-SVR				
Input-set	Transf	gamma	r	devA
ss3	unsquashed	1	0.73	0.76
	squashed	1	0.80	0.63
ss8	unsquashed	1	0.72	0.78
	squashed	1	0.83	0.63

Tab. 3.3: Results for ϵ-SVR using transformed or untransformed target values

and (b)). This means an amino acid is defined to be in contact with another amino acid if any of the carbons in their backbones are within a radius of 4.5 Å. This constraint is always fulfilled for each amino acid with itself and with its direct neighbor. These trivial contacts hold no information and weakens the impact of the interesting contacts. However, as Figure 3.1(c) shows, the profile calculated with a map containing the trivial contacts match the SCHEMA profile calculated from the online tool.

Figure 3.2 shows four example profiles calculated from the online tool and the program developed here.

3.2.2 Transformation

For both SVR and neural networks, preliminary studies showed that using the SCHEMA score directly as target led to slightly worse results than a bounded version (see Table 3.3). Several

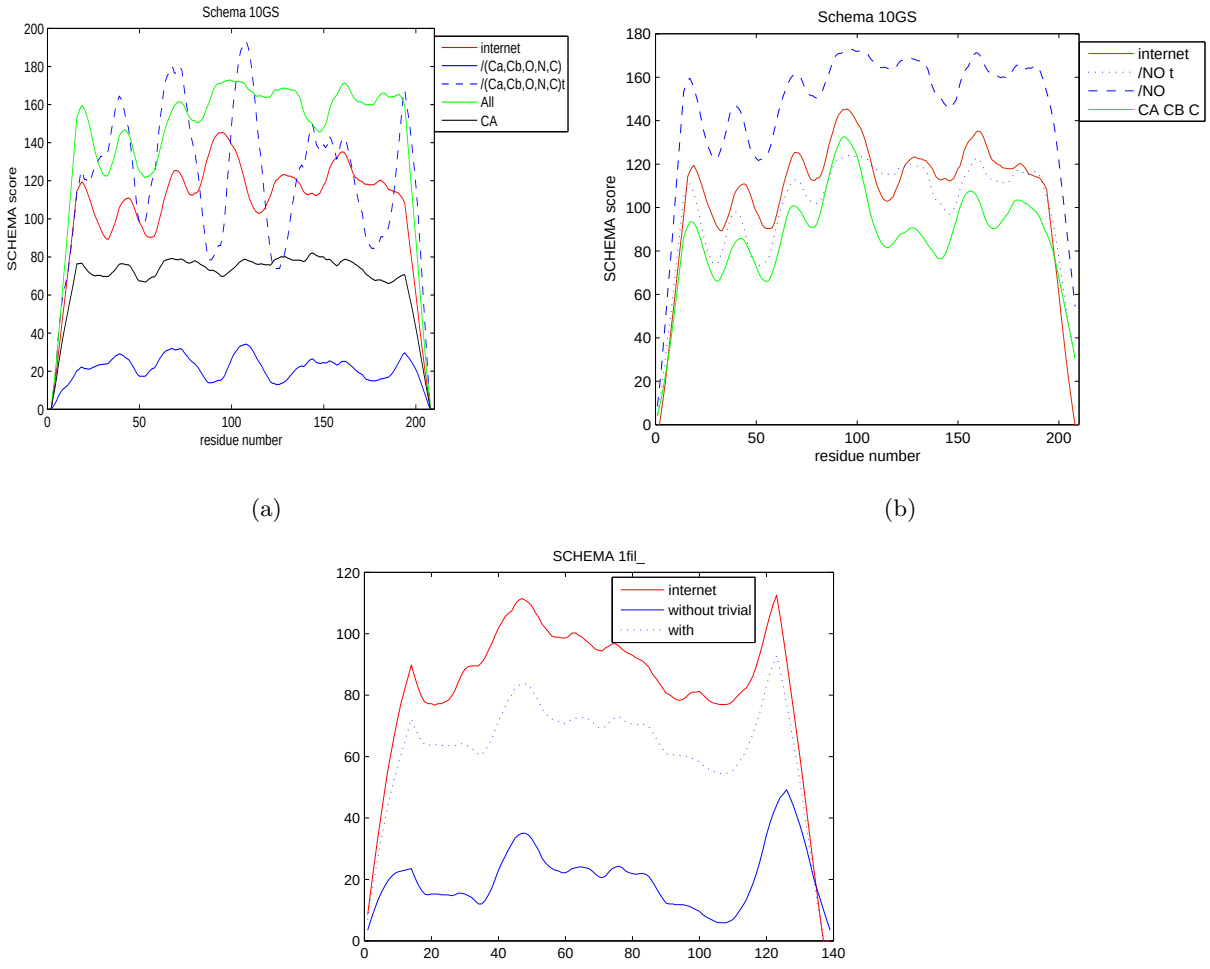


Fig. 3.1: Resulting SCHEMA profile for different methods of producing a Contact Map along with the profile from the online source. t means the profile was linearly transformed for a better comparison. (a) Green is the resulting score, when the pairwise distance between all atoms given in the PDB file are used to determine the distance between two amino acids; black when only the coordinates from the C-α atom from each amino acid is used; blue when only the atoms from the side-chain and not the backbone is used. (b) blue all atoms except NO; green C-α, C-β and C atoms. (c) resulting profiles if a amino acid is defined to be in contact with its neighbor and itself (with trivials) or if the contact Map holds only the real contacts (without trivials).

The profile calculated from the map build of C-α, C-β and C seems to fit best the score from the internet. Same for the map containing trivial contacts.

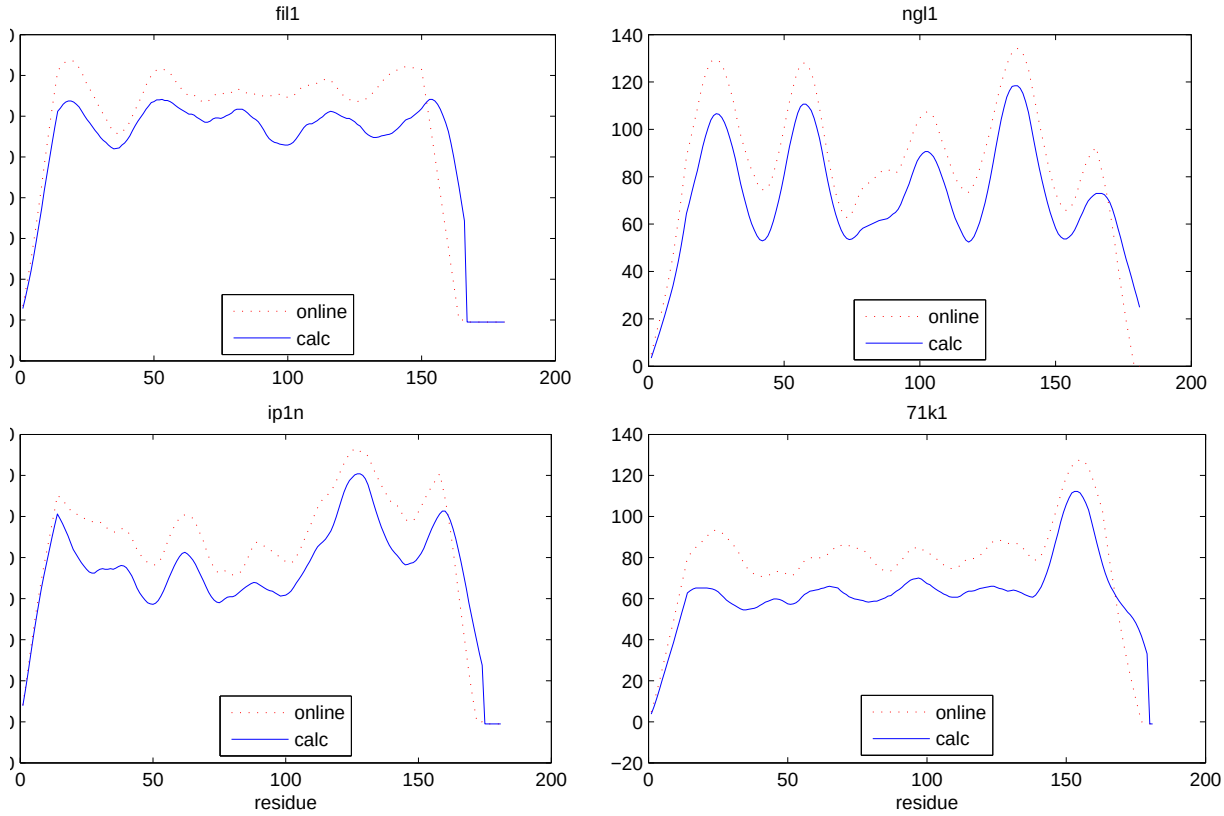


Fig. 3.2: SCHEMA profile from the online tool along with the here calculated SCHEMA profile

transformation functions were surveyed: a simple linear transformation, a zero-one bounded logistic function:

$$S' = \frac{1}{1 + exp(-\frac{X}{n})} \tag{3.2}$$

(where n is a normalization constant)

and a zero-one bounded tanh function:

$$S' = tanh\left(\frac{S}{n}\right) \tag{3.3}$$

where n is a normalization constant.

Two normalization constants were surveyed. The mode, which is the value that occurs most frequently (see Figure 3.3(a)) and the absolute highest value that can occur for an average sized protein (200 bp): if all amino acids are in contact. The mode was 228 and the highest absolute value was 1274.

As shown in Figure 3.3, the linear transformation (blue, solid) would represent the SCHEMA score

best, because it does not change the characteristics in the score function and uses the rage between 0 and 1 best. However, the linear transformation is not zero-one bounded and can cause problems if a new protein produces scores that are greater than 1 after the transformation. The closest zero-one bounded transformation was used to circumvent this problem: tanh with $\frac{1067}{2}$ as normalization (solid red).

The same normalization constant was used for the Nh3d data set as well, because it is unlikely that the proteins produces SCHEMA scores that saturate the tanh function.

3.2.3 The input sequence data

The study evaluates three non-exclusive means for presenting the protein sequence data.

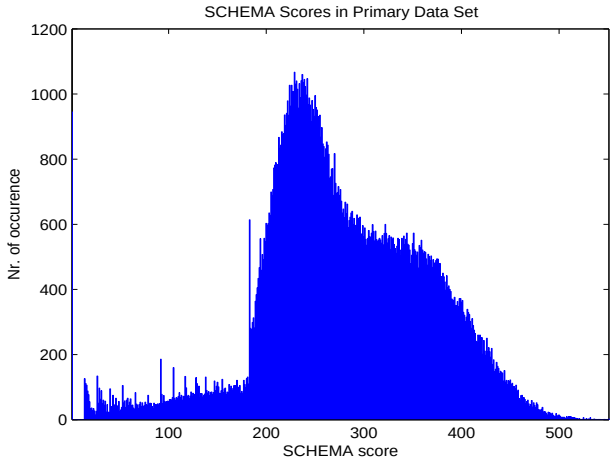
The *primary structure* (ps) is encoded numerically by using so-called PSI-BLAST profiles. Profiles are generated by performing an iterated PSI-BLAST search (three passes against Genbank's non-redundant protein data set). Each sequence is broken down into separate positions. Each position is encoded by a 20-element vector (see Table 3.4). Each element in the vector corresponds to a specific amino acid and its value essentially reflects how often the amino acid appears in this and (determined by PSI-BLAST) very similar sequence positions. Profiles are generally thought to reflect evolutionary information and have been shown to be superior to other means of numerically encoding amino acid information for structure-related prediction [13, 67].

residue nr.	AA	SCHEMA score	PSI-BLAST encoding of the AA
1	M	43	-0.1 -0.2 -0.3 -0.4 -0.2 -0.1 -0.3 -0.3 -0.2 0.1 0.3 -0.2 ...

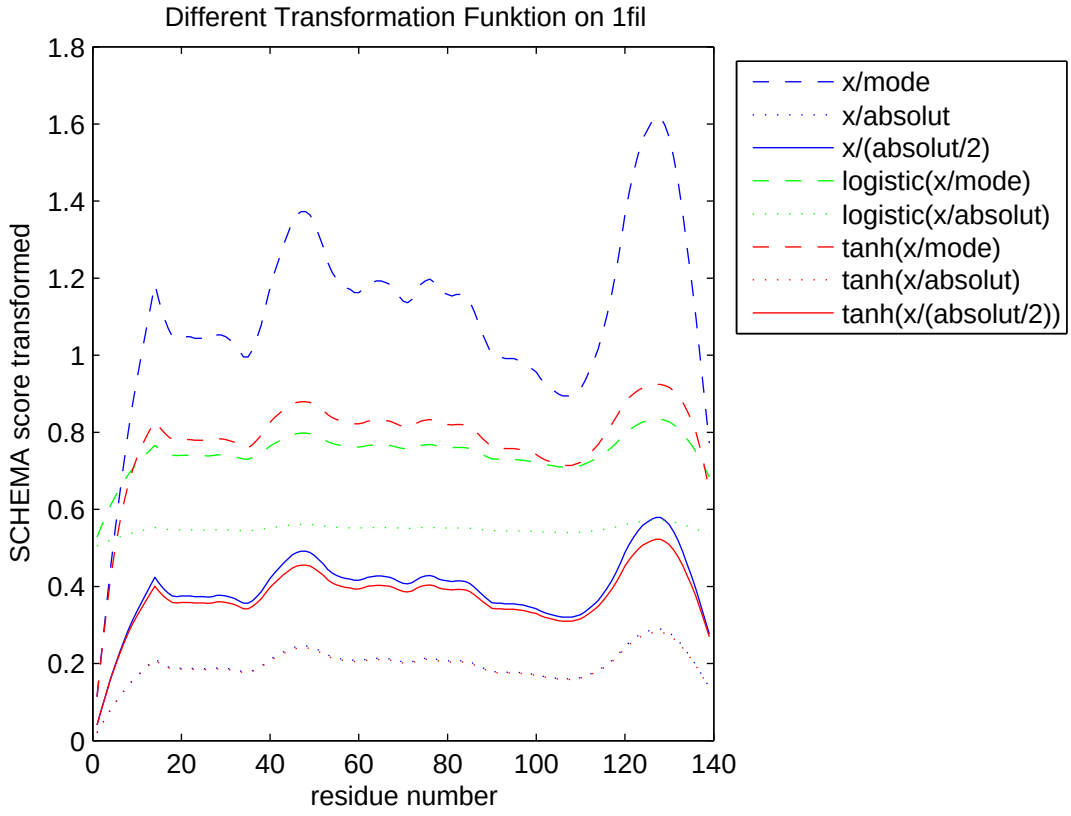
Tab. 3.4: Data provided to the SCHEMA score predictor (Sequence in PSI-BLAST encoding)

SCHEMA scores takes the contact map of the protein into account, since this is a very complex information to predict from the sequence the sequence as input might not be sufficient to successfully predict the SCHEMA score. Different input information were tried, such as *secondary structure*, *secondary structure* with *contact numbers* or solvent accessibility score. The predictors used reached mainly accuracies that are on par with other state-of-the-art predictors and access to the programs was provided.

The *secondary structure* of each residue – either three-state (ss3) or eight-state (ss8) [43] – can be predicted from the sequence data. The Continuum Secondary Structure Predictor introduced by Bodén *et al.* was used [15]. The predictor produces a probability for each *secondary structure*



(a)



(b)

Fig. 3.3: Resulting SCHEMA profile for different methods of transforming the score to values between 0 and 1. (a) Histogram of the SCHEMA score values appearing in the primary data set. (b)Transformation of the SCHEMA score by the different transformation fuctions

state (both for three-state and eight-state). The *secondary structure* model is based on a recently proposed scheme by Andersen *et al.* [50] to more accurately represent caps on regular structures and structural ambivalence in flexible structures. At an accuracy of 0.47, as measured by Kullback-Leibler divergence from the 3-class distribution amongst NMR models, cascaded probabilistic neural networks produce the most accurate continuum *secondary structure* [15]. The classification accuracy achieved by thresholding this probabilistic predictor is, at $Q_3 = 77.3$, on par with standard categorical *secondary structure* predictors.

The predicted probabilities of the continuum *secondary structure* represent the residue when the sequence is presented to the model. The *secondary structure* encoding of a sequence position is thus considerably shorter (three or eight values) than the profile encoding (see Table 3.5).

residue nr.	AA	SCHEMA score	Secondary Structure
1	M	43	0.016 0.018 0.964

Tab. 3.5: Data provided to the Contact Number predictor

Kabakçiogul suggested that information from *secondary structure* in combination with *contact number* describes a protein uniquely [63]. Thus, this work investigates the residue *contact number* (co) as additional input feature to the predictor.

The goal is to predict the SCHEMA score from the primary structure directly, thus, the *contact numbers* used must as well be predicted. Pollastri *et al.* report on a method that is able to derive *contact numbers* [16]. A SVR Contact Number Predictor [17] was used which predicts the contact number for each residue from the primary structure with an correlation coefficient of 0.70. The *contact numbers* are normalized by the following equation:

$$Co' = \frac{Co - \langle Co \rangle}{\sqrt{(Co - \langle Co \rangle)^2}} \tag{3.4}$$

where $\langle \cdot \rangle$ is the mean.

The ability of a third structural feature as input to the SCHEMA score was examined. The *solvent accessibility* score holds information of how exposed the residue is to the solvent. Unlike Curago, the predictions here aims directly to determine the continuous score from the primary structure instead of the binary state of buried or exposed. The *solvent accessibility* predictor used had a correlation coefficient of 0.64. Solvent accessibility score used as input feature to the SCHEMA score predictor was transformed by the following equation:

$$SolvAcc' = tanh(\frac{SolvAcc}{60}) \tag{3.5}$$

3.3 Predictors

Two major types of machine learning algorithms were evaluated, namely Neural Networks and Support Vector Regression. The choice of techniques is supported by the general observation that these two types of algorithms have repeatedly been found superior for relevant prediction problems (e.g. *secondary structure* prediction [13, 14, 15], contact number and solvent accessibility prediction [16, 17]). The model configuration were in accordance with previous studies of predicting structural features. After initial experiments the optimal window size was found to be 15 residues: the residue for which the SCHEMA score is predicted and then 7+7 residues immediately upstream and downstream, respectively. This observation is consistent with previously reported parameters for most *contact number* and *secondary structure* predictors [13, 14, 16]. The extended surveys were therefore undertaken with this windows size for all models. Algorithm-specific parameter values are provided in the following.

3.3.1 Neural Networks

The Feed Forward Neural Network (FFNN) is trained and evaluated on the data set. The number of input nodes of the FFNN depends on the input encoding (3, 8, or 20). The network is standardly trained using gradient descent to minimise the error as measured on the single output node. The learning rate is $\eta = 0.001$ which had proven to be a good compromise between accuracy and timesteps needed to reach the minimum for a wide range of reported neural network approaches. A variety of hidden node numbers h (including not using a hidden layer at all) are trialled. A sigmoidal output function is used because regression tend to be more accurate on a bounded interval than with a liner output function.

Furthermore, a Bidirectional Recurrent Neural Network (BRNN) is trained and evaluated on exactly the same data. BRNNs have previously been found to be superior to FFNN for both contact number and *secondary structure* prediction [3, 55]. The basic configuration of the FFNN was again used for the BRNN. However, the BRNN requires a modified training procedure that works with the use of upstream and downstream input "wheels" [68]. Wheels allow a much greater number of residues to be part of the input, without introducing a major increase in the number of weights to be adapted by the training algorithm. The number of hidden nodes in each of the wheels is set to seven in all tests.

For all neural networks, training data is presented in batches of 100 windows before the weights

Linear	$k(x,y) = (x \cdot y)$
Gaussian RBF	$k(x,y) = exp(\frac{-\gamma\|x-y\|^d}{c})$
Sigmoidal	$k(x,y) = tanh(\gamma \cdot x \cdot y + c)$

Tab. 3.6: Kernel functions used

are changed. A total of 40,000 sequences were presented in random order before the training was stopped. In preliminary studies this number was seen as sufficient for convergence (see Figure 4.3).

3.3.2 Support Vector Regression

Recent findings suggest that Support Vector Regression (SVR) exceeds the accuracy reached by many neural networks [69, 17]. Two SVR methods were surveyed ϵ-SVR and ν-SVR which were provided with the same protein data set as for the neural networks. The standard stopping criterion was used and $C = 0.5$ (ensuring a medium balance between penalizing misclassifications and maximizing the size of the decision margin). Three different standard kernel functions are tested (Table 3.6, with $\gamma = 1, c = 1$ and $d = 3$). The LIBSVM implementation of the optimisation procedures was used [70].

3.4 Ensemble

Another way to increase the accuracy, besides using different machine learning approaches or parameter tuning, is to combine the output of several individually trained methods. This concept is called ensembling and has in some cases proven to be significantly more accurate than any of the single methods in the ensemble [71].

This work investigates in 2 ensemble types: The first ensemble has the output of a BRNN and a FFNN as basis and the second the output of BRNN and ϵ-SVR. Therefore the method reaching the highest accuracy (see result section 4) is combined with both alternative approaches.

Several approaches can be used to derive the final output out of the component outputs. Many researchers have demonstrated that an effective combining scheme is to simply average the predic-

tions of the network [71]. Thus, this work investigates in this simple method to detect the ability in improving the prediction beyond the accuracy of the best method surveyed so far.

3.5 Testing

Simulations with the used models showed that the differences between 10-fold and 2-fold crossvalidation are negligible (see result Table 4.2 and Table 4.3). Thus, to minimize the computational time required, 2-fold crossvalidation was used for testing the models.

Performance Measure

Two performance measures were employed. The first one is the correlation coefficient r between the calculated SCHEMA score t_i and the predicted value p_i where the index i represents the position in the sequence.

$$r = \frac{\langle (t_i - \langle t_i \rangle) \cdot (p_i - \langle p_i \rangle) \rangle}{\sqrt{\langle (t_i - \langle t_i \rangle) \rangle} \cdot \sqrt{\langle (p_i - \langle p_i \rangle) \rangle}} \tag{3.6}$$

where $\langle \cdot \rangle$ is the mean. Ideal performance means that t_i and p_i are perfectly and positively correlated $r = 1$.

The second measure $DevA$ is the Root Mean Square Error (RMSE) normalized by the standard deviation of the target.

$$DevA = \frac{\sqrt{\langle (p_i - t_i)^2 \rangle}}{\sqrt{\langle (t_i - \langle t_i \rangle)^2 \rangle}} \tag{3.7}$$

Ideal performance means that t_i and p_i are identical hence $DevA = 0$.

Both measures are defined for a single protein chain. All reported result values are averages over all chains (when they appear as test cases).

3.6 Summary of the Approach

The first attempt will be the prediction from primary structure. Additionally the prediction from predicted structural features (*secondary structure*, *contact numbers* or solvent accessibility in combiation with *secondary structure*) will be surveyed in order to derive the SCHEMA profile. This in turn can be used to determine the discreet recombination sites (Figure 3.4).

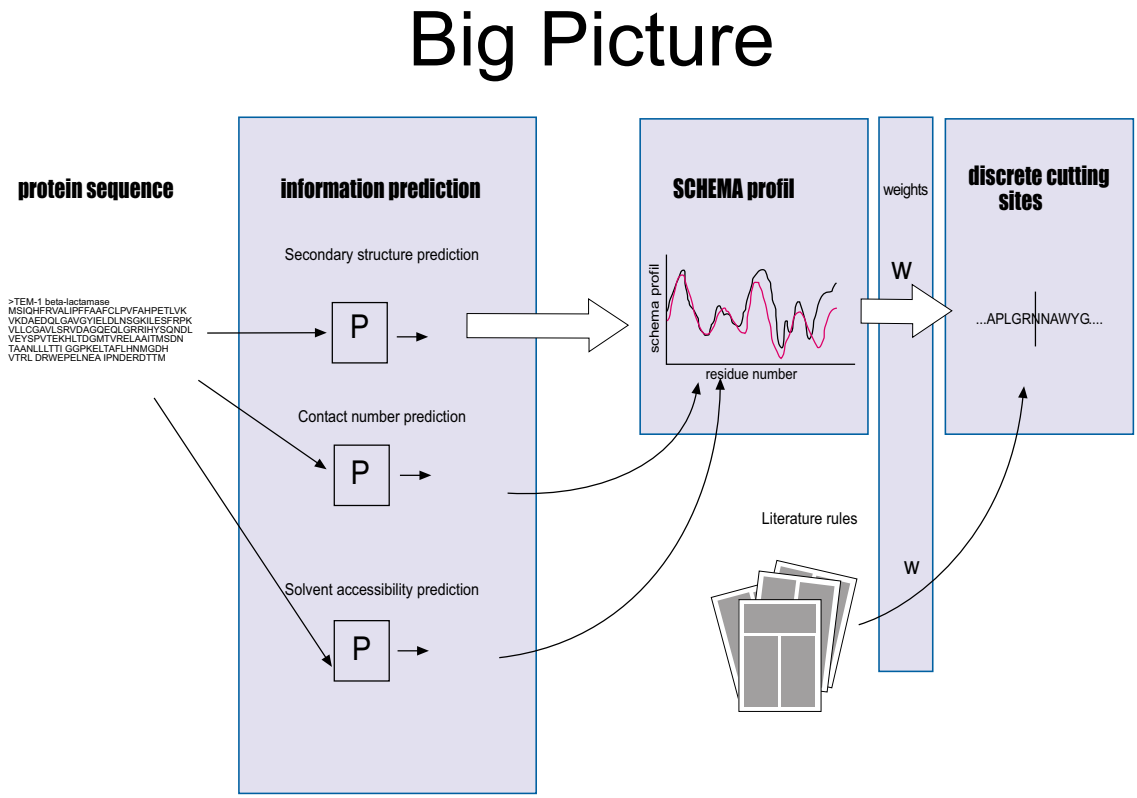


Fig. 3.4: Basic idea of information management in a future recombination site predictor. The protein sequence will be the only information to the predictor complex. From this sequence the psi-blast encoding will be derived to provide the basis for the *secondary structure* prediction. From the predicted *secondary structure* the SCHEMA profile will be predicted. The minima of this profile suggest possible cutting sites. The suggested cutting sites can then be evaluated or extended by other methods such as an expert system derived from rules found in wet-lab experiments

4. RESULT AND DISCUSSION

4.1 The primary structure as input

The initial set of simulations show that machine learning algorithms are unable to predict the SCHEMA score directly from the primary structure (see Table 4.2 and Table 4.3). This result is not surprising, as mentioned earlier, the contact map which is used for the SCHEMA calculation is difficult to predict (almost as complex as tertiary structure itself) from the sequences and so is the SCHEMA score. The alternative, learned from tertiary structure prediction, will be discussed in the following sections.

4.2 Classification

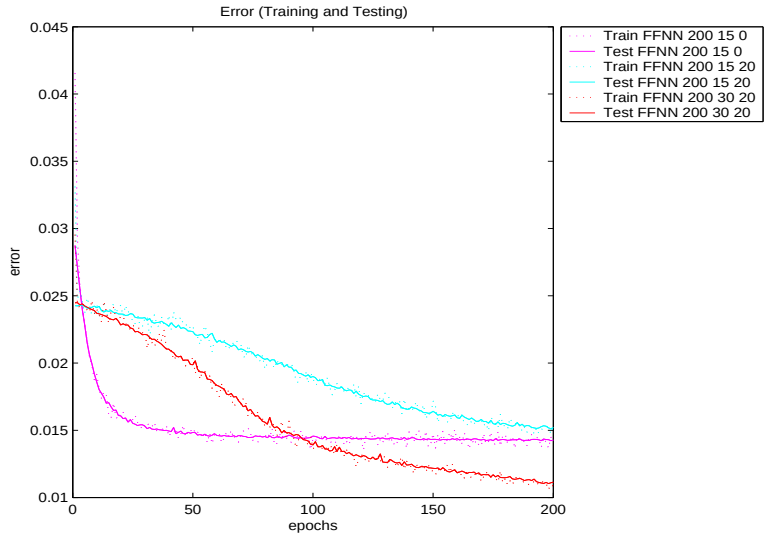
Since predicting from primary structure failed. An attempt was made to transform the SCHEMA score prediction to a classification problem. Similar approaches are known from solvent accessibility prediction, where not the exact score is predicted but a number class covering a larger interval or even a binary classification [66].

The continuous profile has to be divided into discreet subsections that can later on be predicted. Two transformations of the profile were surveyed: first classification according to the gradient Box 4.1 and second according to the SCHEMA score itself, where the value range of the SCHEMA score was simply divided into sections. The resulting segmentation is shown in Figure 4.2

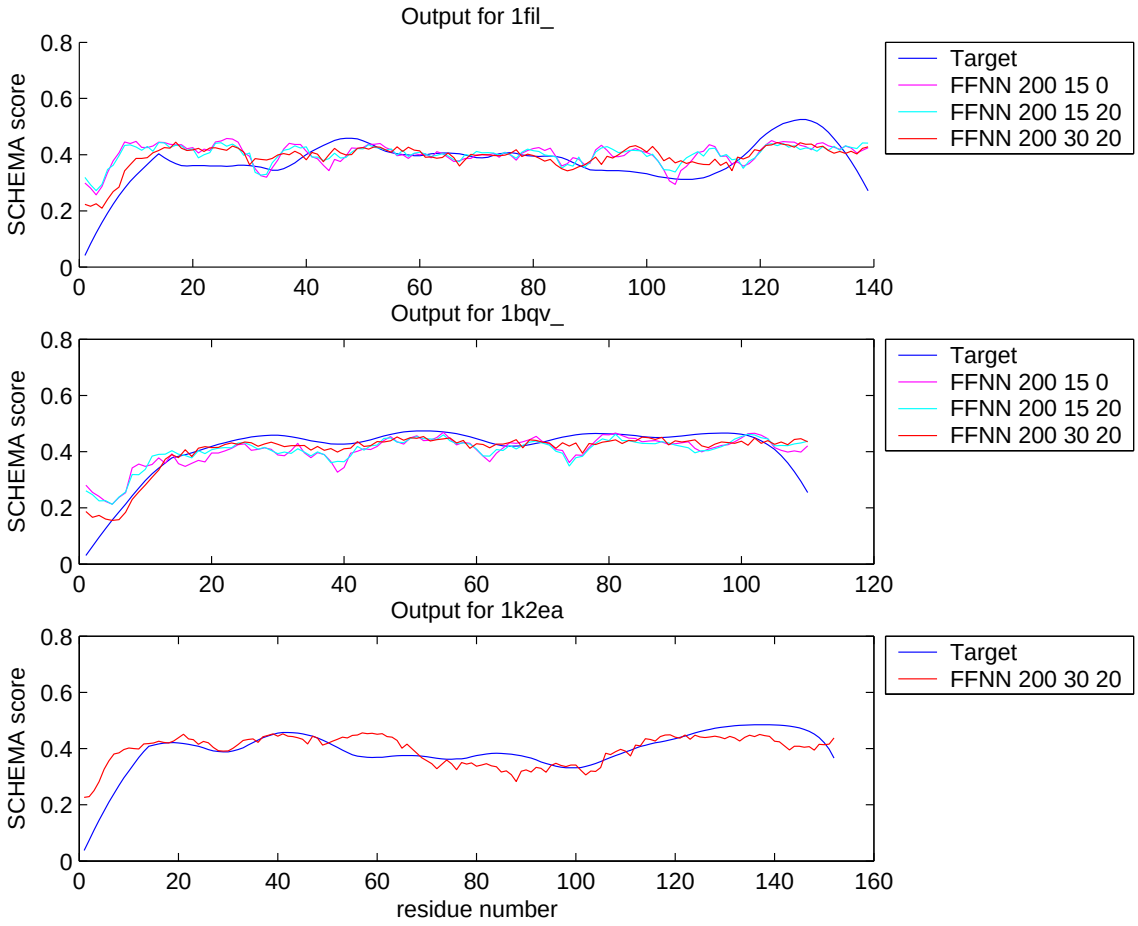
This work does not follow the classification path further because dividing the profile into sections causes too much loss of information and the recombination site detection is not possible anymore.

4.3 The secondary structure as input

The results when *secondary structure* is used as input for the two neural network architectures are shown in Table 4.2 and Table 4.3. The results when *secondary structure* is used as input for the



(a)



(b)

Fig. 4.1: Predicted SCHEMA profile along with the target profile. The predictors were provided with primary structure in Psi-Blast encoding

Gradient
no gradient
$v1 - v2 < eps \;\; \&\& \;\; v1 - v2 > -eps$
positive gradient
$v1 - v2 < -eps$
negativ gradient
$v1 - v2 > eps$

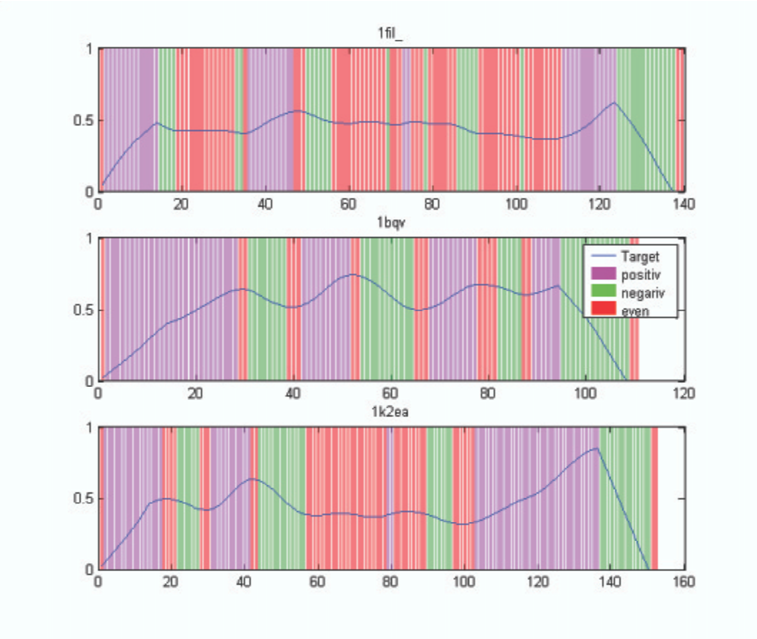
Tab. 4.1: pseudocode for gradient detection, eps was chosen to be 4

support vector regression algorithm are shown in Table 4.4 and Table 4.5.

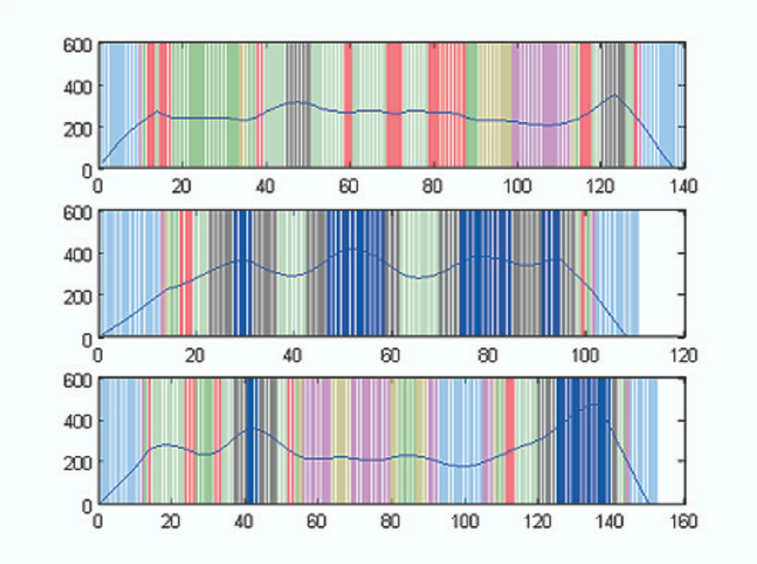
Notable is that a single-layer neural network performs surprisingly well on the ss3-input data with $r = 0.86$ and, for ss8, even better than a multi-layer neural network with 20 or 40 hidden units. As expected, the BRNN outperforms the FFNN with $r = 0.88$ on the ss3 data (and slightly worse for the ss8 data). This result can not be significantly improved by increasing the training time. As shown in Figure 4.3, the error function has flattened at 40,000 presented sequences. Also shown in Figure 4.3, there is no overfitting since the testing error did not rise above the training error. The predicted outputs of the BRNN for four sequences are shown in Figure 4.4 and 4.5.

Neither SVR optimization algorithm performs better than the bi-directional neural network. ν-SVR reaches $r = 0.85$ on ss3 and $r = 0.87$ on ss8 both with the simple linear kernel.

The tests suggest that neural networks are usually better than SVR at predicting of the SCHEMA score. However, the low number of tests and the small differences prohibit us from making any general claims regarding how the algorithms compare. Increasing the performance of SVR could be a matter of choosing a more suitable kernel, such as string kernel e.g. local alignment kernel [72]. The good performance of the linear single-layer neural network and the linear kernel SVR indicates a linear correlation between the structural features and the SCHEMA score. The better performance of BRNNs over FFNNs indicates that there are useful dependencies in the data beyond the window size of 15 residues. The SCHEMA equation itself is window-based and takes therefore only local interactions into account. These local interactions, however, are influenced by the global structure of the protein and therefore by long term dependencies.



(a)



(b)

Fig. 4.2: SCHEMA profile transformation in a classification problem (a) according to the gradient (b) by division of the SCHEMA score value range

4.4 Nh3d data set

As previously mentioned, the Nh3d data set was intended to verify the accuracy determined for the extensive examined primary data set. The results are shown in Table 4.6 and Table 4.7. The Predictor performs slightly worse than on the primary data set. However, this was expected since

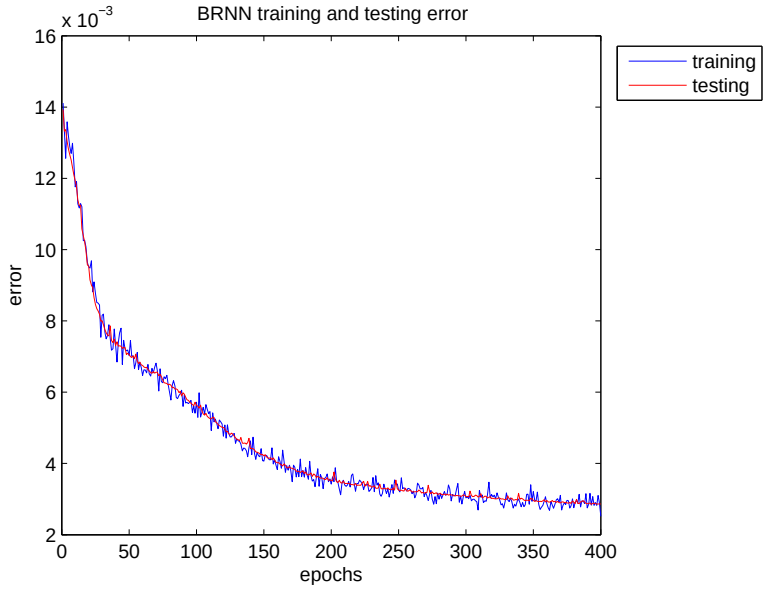


Fig. 4.3: Testing and training error for BRNN 2-fold crossvalidation, 40,000 sequences, 7+1+7 residue window

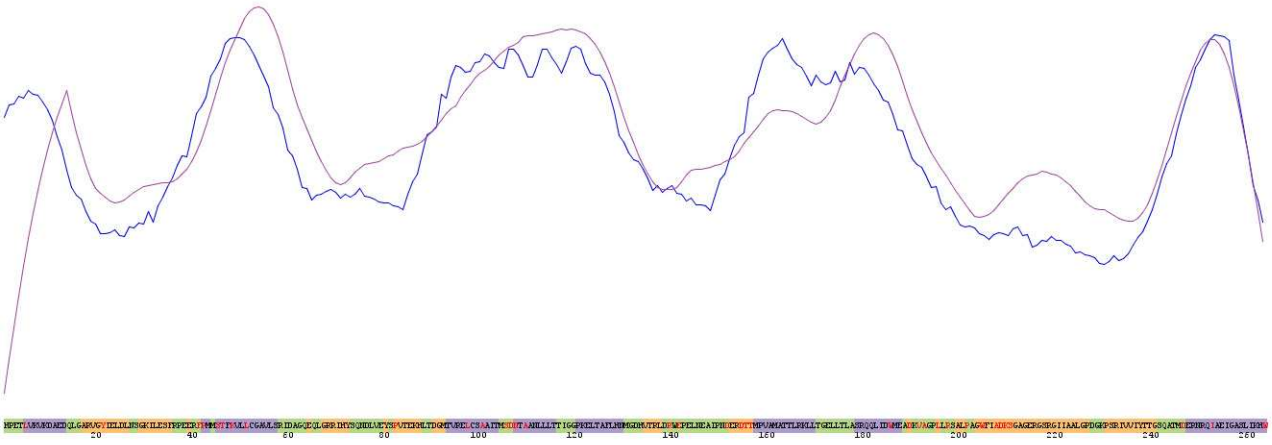


Fig. 4.4: Secondary structure of Q4AE67 along with the predicted (blue) and the calculated (lavender) SCHEMA score. α-helix is light lavender, β-sheet is yellow and coil is light green

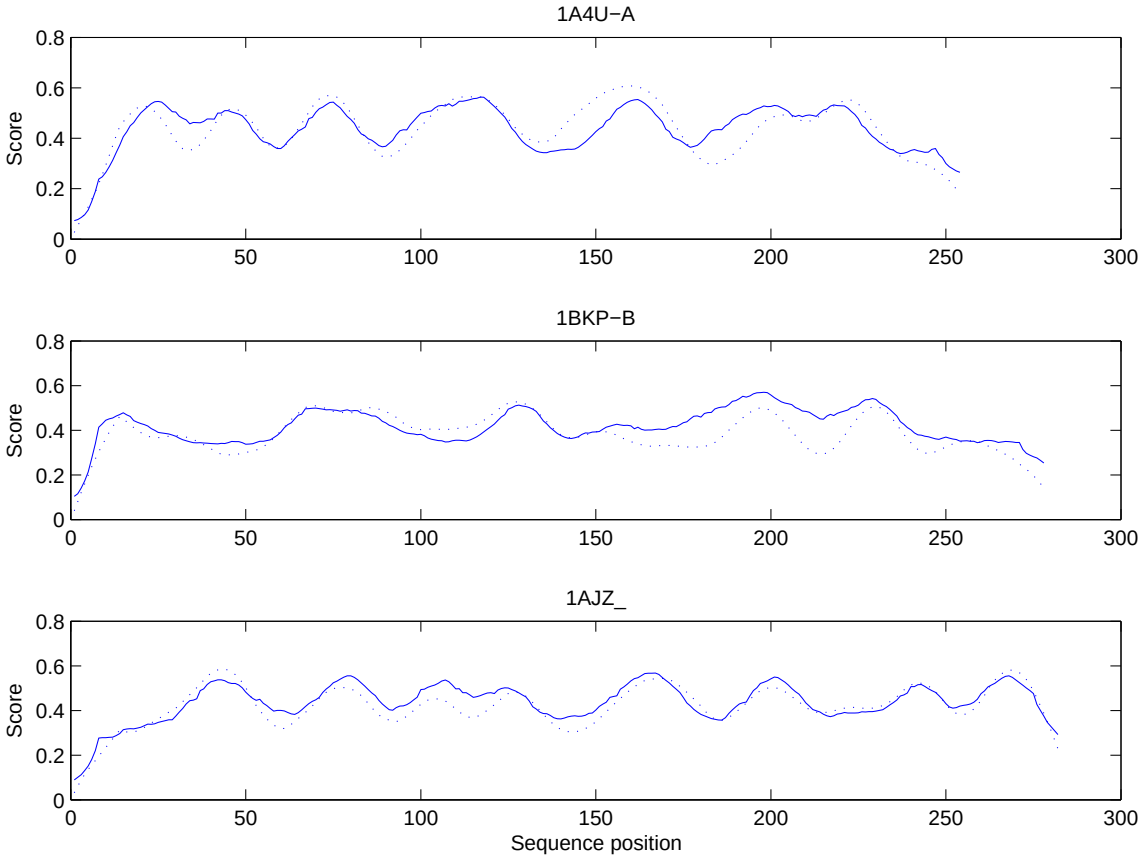


Fig. 4.5: Examples of predictions produced by the BRNN with ss3 input ($h = 7 + 7$, $w = 15$, $r = 0.88$). The solid line represents the actual SCHEMA score and the dashed line represents the predicted score.

FFNN						
Inputset	epochs	h	2-fold		10-fold	
			r	devA	r	devA
ps	20,000	0	0.56	0.96	0.56	0.95
		20	0.54	0.99	0.53	0.97
		40	0.54	0.98	0.54	0.98
	40,000	0	0.56	0.95		
		20	0.56	0.96		
		40	0.56	0.95		
ss3	20,000	0	0.85	0.60	0.85	0.59
		20	0.82	0.65	0.82	0.66
		40	0.83	0.63	0.83	0.64
	40,000	0	0.86	0.58		
		20	0.86	0.57		
		40	0.86	0.58		
ss8	20,000	0	0.85	0.60	0.85	0.60
		20	0.80	0.68	0.81	0.68
		40	0.81	0.67	0.81	0.67
	40,000	0	0.86	0.57		
		20	0.85	0.60		
		40	0.86	0.58		

Tab. 4.2: The performance of FFNN (using 15-residue input window, and training for 20,000/40,000 sequences).

the proteins in the Nh3d data set share less similarities. Training and testing set is therefore more diverse which means there is less incidents where the predictor just recalls previously learned.

4.5 The contact number as input

The reported results suggest that there is a direct relation between some structural features and the SCHEMA score. Hence, adding another structural feature to the input may assist further in improving the prediction accuracy. To test this hypothesis the *contact number* for each residue in the data set was predicted and used as an additional input feature for the SCHEMA score predictor.

BRNN						
Inputset	epochs	h	2-fold		10-fold	
			r	devA	r	devA
ps	20,000	7+7	0.56	0.98	0.56	0.99
	40,000	7+7	0.66	0.90		
ss3	20,000	7+7	0.87	0.57	0.87	0.95
	40,000	7+7	0.88	0.52		
ss8	20,000	7+7	0.88	0.58	0.87	0.58
	40,000	7+7	0.87	0.55		

Tab. 4.3: The performance of BRNN (using a 7+1+7 residue input window, and training for 20,000/40,000 sequences).

ϵ-SVR			
Inputset	kernel	r	devA
ss3	linear	0.82	0.63
	hbf	0.80	0.65
	sigm	0.07	1669.8
ss8	linear	0.85	0.60
	hbf	0.83	0.64

Tab. 4.4: The performance of ϵ-SVR (using 2-fold crossvalidation, and a *secondary structure* 15-residue input window).

As shown in Table 4.8, Table 4.9 and Table 4.10, the additional information seems to make no contribution to the prediction accuracy in neither of the models. The lack of improvement is not explained by inaccuracy of the *contact number* predictor. Several tests were carried out with the FFNN trained on *secondary structure* and *contact number* determined directly from the contact map (PDB description). The difference in using this observed *contact number* and the predicted *contact number* is negligible (Table 4.8), which strengthens the conclusion that additional *contact numbers* are not aiding in improving the accuracy further.

Recalling that the SCHEMA score was calculated with a window approach may explain this observation: *contact numbers* can be divided into local connections and global connections. *Secondary structure* essentially covers the information content of local connections. Since the SCHEMA score is calculated using a window we suggest that the additional information about global connections

ν-SVR			
Inputset	kernel	r	devA
ss3	hbf	0.83	0.62
ss8	linear	0.84	0.60
	hbf	0.85	0.58

Tab. 4.5: The performance of ν-SVR (using 2-fold crossvalidation, and a *secondary structure* 15-residue input window).

FFNN on Nh3d				
Inputset	epochs	h	r	devA
ss3	40,000	0	0.85	0.59
		20	0.85	0.58
		40	0.85	0.57
ss8	40,000	0	0.85	0.57
		20	0.84	0.60
		40		

Tab. 4.6: The performance of FFNN on Nh3d (using 2-fold crossvalidation, a *secondary structure* 15-residue input window, and training for 40,000 sequences).

is not aiding since the SCHEMA score does not take these connections into consideration.

4.6 Solvent accessibility as input

Unlike *contact numbers solvent accessability* information has a major negative impact on the prediction (see Table 4.11). As mentioned earlier the prediction accuracy for the solvent accessibility score was 0.64 which is certainly one reason for the poor performance. The main reason, however, might be that the solvent accessibility score used here holds little and even confusing information for the predictor. Another representation of solvent accessibility information would be required.

Additional Information as input needs to be investigated further to reach a conclusion about their suitability for the SCHEMA score prediction.

BRNN on Nh3d			
Inputset	h	r	devA
ss3	7+7	0.86	0.54
ss8	7+7		

Tab. 4.7: The performance of BRNN on Nh3d (using 2-fold crossvalidation, a *secondary structure* 7+1+7 residue input window, and training for 40,000 sequences).

FFNN					
Inputset	h	predicted CO		observed CO	
		r	devA	r	devA
ss3 co	0	0.86	0.57	0.86	0.57
	20	0.86	0.57	0.85	0.57
	40	0.86	0.57	0.86	0.57
ss8 co	0	0.86	0.57	0.86	0.56
	20	0.85	0.59	0.85	0.59
	40	0.85	0.59	0.85	0.60

Tab. 4.8: The performance of FFNN using the *secondary structure* and predicted/observed *contact number* as input (2-fold crossvalidation, a 15-residue input window, and trained for 40,000 sequences).

4.7 Ensemble

The accuracy from the ensemble is not higher than the accuracy of the best component alone. This finding is in consistency with the observation from Rost, namely that for protein data a single predictor often produces a higher accuracy than a simple ensemble [19].

However, more sophisticated methods, such as boosting [71] might be able to reach better performance nevertheless. The idea behind boosting is to reduce training error by retraining on difficult

BRNN			
Inputset	h	r	devA
ss3 co	15	0.88	0.52
ss8 co	15	0.87	0.58

Tab. 4.9: The performance of BRNN using the *secondary structure* and predicted *contact number* as input (2-fold crossvalidation, a 7+1+7 residue input window, and trained for 40,000 sequences).

ϵ-SVR				
Inputset	kernel	transf	r	devA
ss3 co	linear	1	0.82	0.63
	hbf	1	0.80	0.66
ss3 co	linear	1	0.85	0.60
	hbf	1	0.71	0.84

Tab. 4.10: The performance of ϵ-SVR using the *secondary structure* and predicted *contact number* as input (2-fold crossvalidation, and a 15-residue input window).

BRNN			
Inputset	h	r	devA
ss3 solvacc	15	0.54	0.92

Tab. 4.11: The performance of BRNN using the *secondary structure* and predicted *contact number* as input (2-fold crossvalidation, a 7+1+7 residue input window, and trained for 40,000 sequences).

data points. Boosting produces a series of training sets, where the next training round uses a sample that contains more data points that were predicted wrong by the previous round.

4.8 Evaluation of the Predicted Profile

4.8.1 Predicted Minima Compared with Target Minima

The most valuable information within the SCHEMA score are according to Arnold *et al.* the minima. It is therefore not imperative to approximate the function exactly as long as the minima are the same. The distance between the positions of the minima in the predicted score and in the target function, holds therefore more information about the suitability of our method for protein design than the correlation coefficient of the whole function.

The minima in the function are identified by a simple algorithm that detects a slope-change. Before applying this algorithm the function were smoothed with a linear kernel the mean of the values within a window. The window size for the target function was $w = 3$. For the predicted function the window size was iteratively increased to avoid a number of minima in P that exceeds the number of minima in T by a factor of 3.

Deriving the distance is not trivial, because the number of minima differs between the predicted

Ensemble						
Ensemble	BRNN		ϵ-SVR		combined	
	r	devA	r	devA	r	devA
BRNN SVR	0.87	0.57	0.80	0.65	0.86	0.60
Ensemble	BRNN		FFNN		combined	
	r	devA	r	devA	r	devA
BRNN FFNN	0.87	0.57	0.85	0.60	0.87	0.56

Tab. 4.12: All predictors were trained on *secondary structure* as input. BRNN (2-fold crossvalidation, a 7+1+7 residue input window, and trained for 20,000 sequences) combined with a ϵ-SVR (2-fold crossvalidation, hbf) and BRNN combined with FFNN (2-fold crossvalidation, a 15 residue window, and trained for 20,000 sequences)).

and the target function. For each minimum in the target function a corresponding minimum in the predicted function has to be identified. This problem can be seen as an optimization task where the corresponding minima in the predicted function has to be chosen in a way that the overall distance is minimized. We choose a dynamic programming approach to solve this optimization-problem.

$$C_{i,j} = Min \begin{cases} C_{i-1,j-1} + abs(P_j - T_i), & \text{if } \langle T_i, P_j \rangle; \\ C_{i,j-1} + 10, & \text{if } \langle -, P_j \rangle; \\ C_{i-1,j} + 10, & \text{if } \langle T_i, - \rangle. \end{cases} \tag{4.1}$$

where $\langle \cdot, \cdot \rangle$ indicates an alignment. The gap-penalty of 10 has proven to be a good measure for the dataset.

The closer evaluation of the best model (BRNN) delivers the following results: The average distance between the position of the predicted and the target minima are 3.42 residues. A scatter plot of the position of minima m_i in the predicted score against the position in the target function for all minima M in the dataset is shown in Figure 4.6.

4.8.2 Predicted Profile compared with biological verified results

After demonstrating the accuracy of the predicted function compared with the target function, the SCHEMA approach itself can now be tested on recombination sites reported in the literature. Testing the SCHEMA approach extensively was previously not possible because in most of the biological experiments the 3D structure is not derived for those proteins mainly generated for a

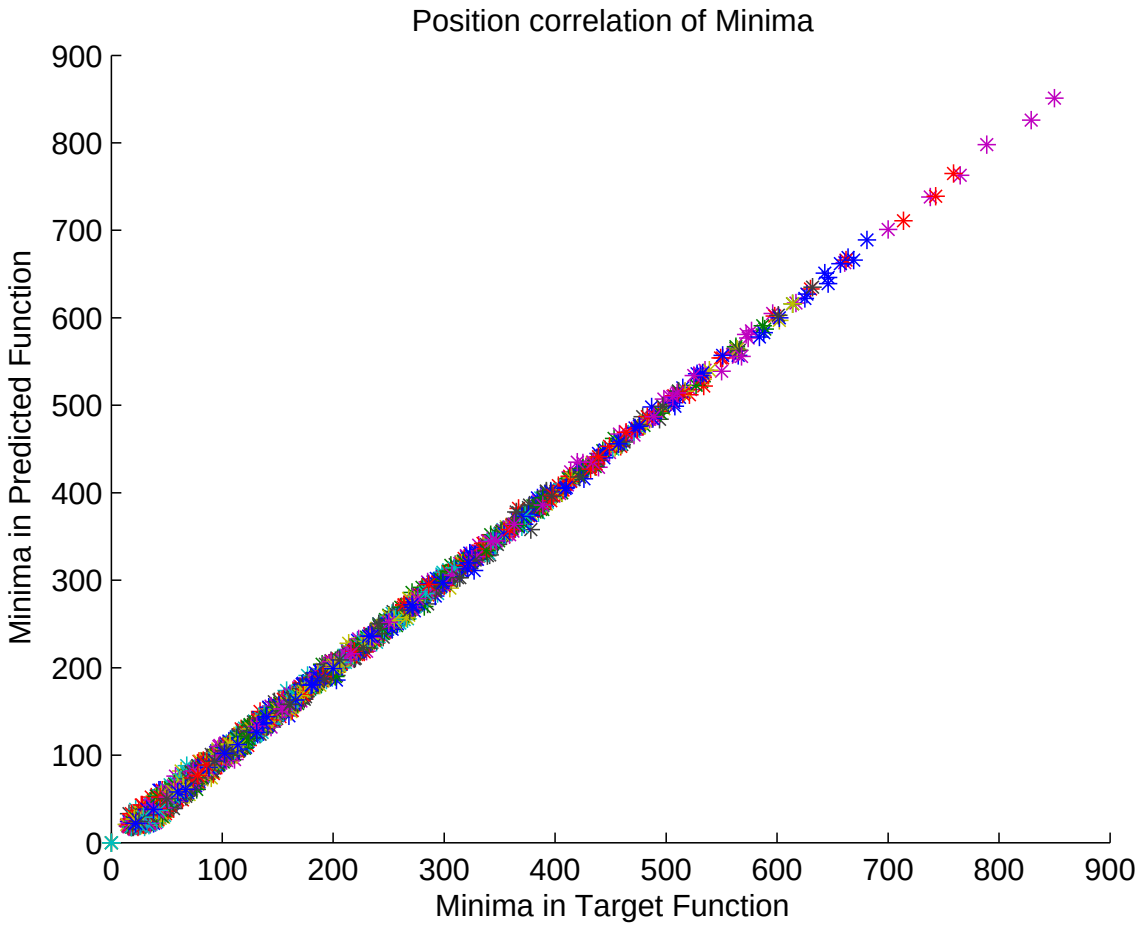


Fig. 4.6: A scatter plot of the position of minima m_i in the predicted score against the position in the target function for all minima M in the dataset. Ideally, the plot forms a perfect diagonal.

Residues Position			
Leu28	Ala132	Trp227	Trp286
Tyr44	Arg162	Ala230	Ala,Glu35
Phe64	Glu164	Asp231	Gly,Ala52
Pro65	Asp174	Lys232	Val,Leu72
Ser68	Asp177	Ser233	Tyr,His103
Thr69	Thr178	Gly234	Ala,Thr133
Lys71	Thr179	Gly239	Pro,Ala181
Leu74	Trp208	Arg241	Leu,Val223
Pro105	Asp212	Gly242	Ala,Gly235
Leu120	Val214	Leu247	Tyr,Phe260
Ala123	Ala215	Gly248	Ile,Val275
Ser128	Leu218	Met268	
Asp129	Arg220	Asn272	
Asn130	Pro224	Ile278	

Tab. 4.13: The residues in *beta*-lactamase that either do not tolerate a substitution or can be replaced by just one other residue. Table from [74]

research purpose.

Critical Amino Acids within the Sequence

Huang *et al.* report on a study that was aimed at determining the amino acid residues that are accepted at each position of the β-Lactamase protein [73]. The TEM-1 gene was mutated exhaustively by random replacement mutagenesis and insertion mutants (three codons at a time) with only one substitution at a time. A residue was found to be accepted if the phenotype showed the ability to confer ampicillin resistance to Escherichia coli. The successful mutants were then gene-sequenced. The results showed that 43 positions in the sequence do not tolerate any kind of substitutions while some other positions were found to accept two different amino acids, and so on (Table 4.13) [74].

This work uses the data reported in Dubey *et al.*'s paper because here the exact sequence is known. They used the TEM-1 beta-lactamase entry, Q4AE67 in UniProtKB/TrEMBL [74].

The SCHEMA score predicted for the given sequence along with the reported critical residues is

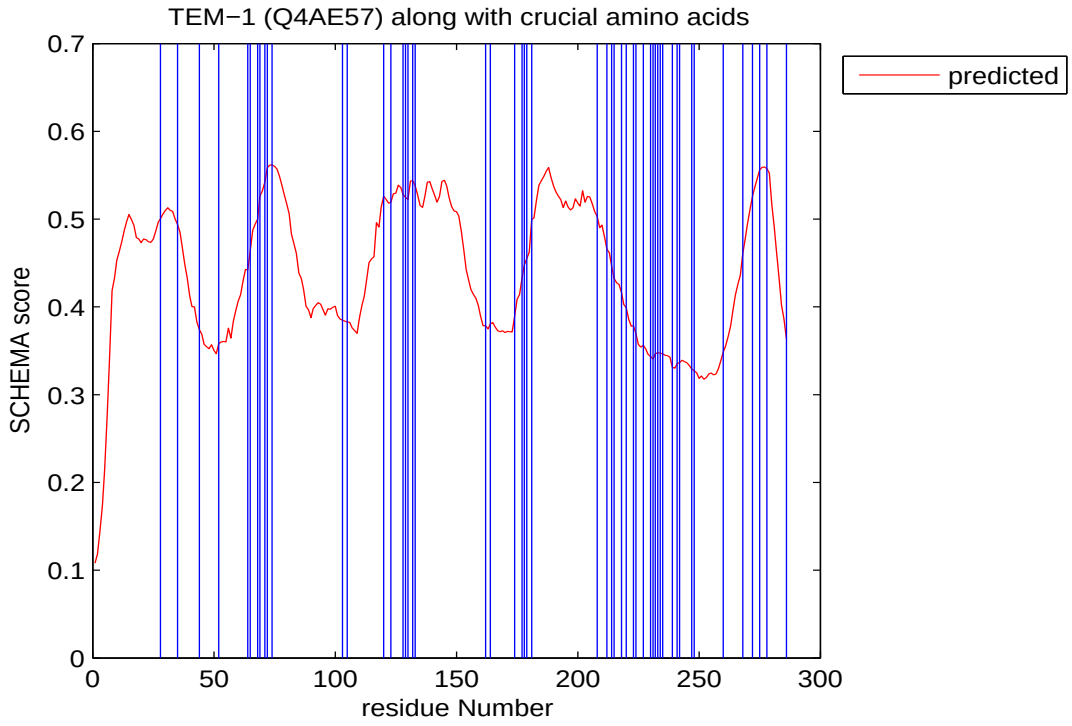


Fig. 4.7: Critical residue position along with the predicted SCHEMA profile

shown in Figure 4.7. Minima in the SCHEMA score indicate possible recombination sites which divide the protein in blocks that are able to fold fairly independently. The critical amino acids are expected to lie within those blocks and have connections (ideally) only with residues within those blocks. The critical positions are clear of the minima. The only exception is position 52 which is an amino acid tolerating the exchange with one other amino acid and therefore not as critical as the ones not tolerating an exchange at all (see Table 4.13).

However, more important than the actual positions of the critical residues are their connections. The tertiary structure was determined from the β-Lactamase protein in PDB, 1ZG4. The sequences were aligned to determine the positions of the critical residues [1]. The global alignment shows that 1ZG4 and Q4AE67 have almost 100% identity, however the first 24 residues are missing in 1ZG4.

The connections of the critical residues would ideally form an independent block whose borders are indicated by minima in the SCHEMA profile. This is obviously not the case as shown in Figure 4.8 a. The SCHEMA score even indicates cutting sites where the critical amino acids have a high connection rate (black triangle in Figure 4.8 a). The reason is that the SCHEMA calculation weights every amino acid equally and is therefore not aware of critical residues. Since there are more

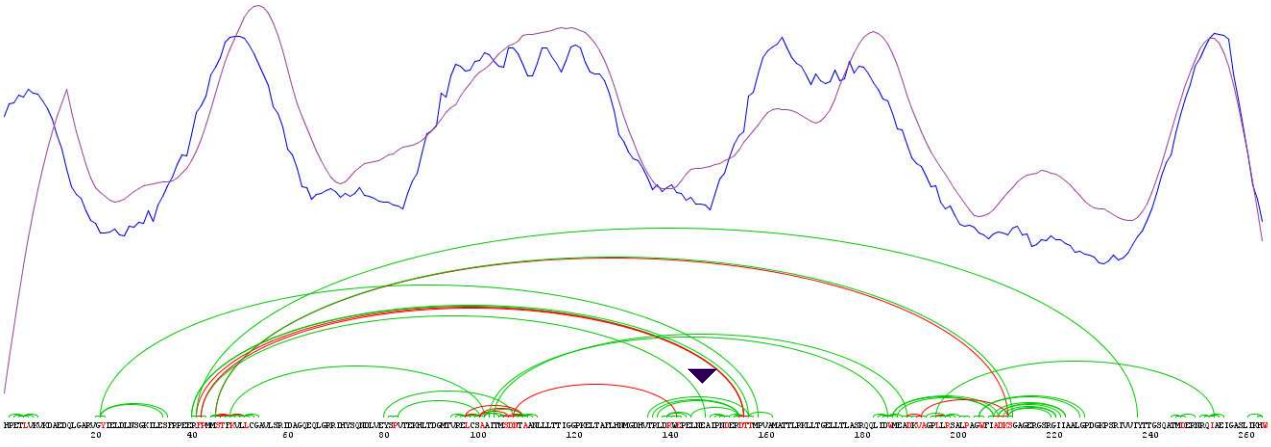
[1] Myers and Miller, CABIOS (1989).4:11-17

connections at other positions, it appears to be favorable for SCHEMA to cut at those positions with the least interruptions overall.

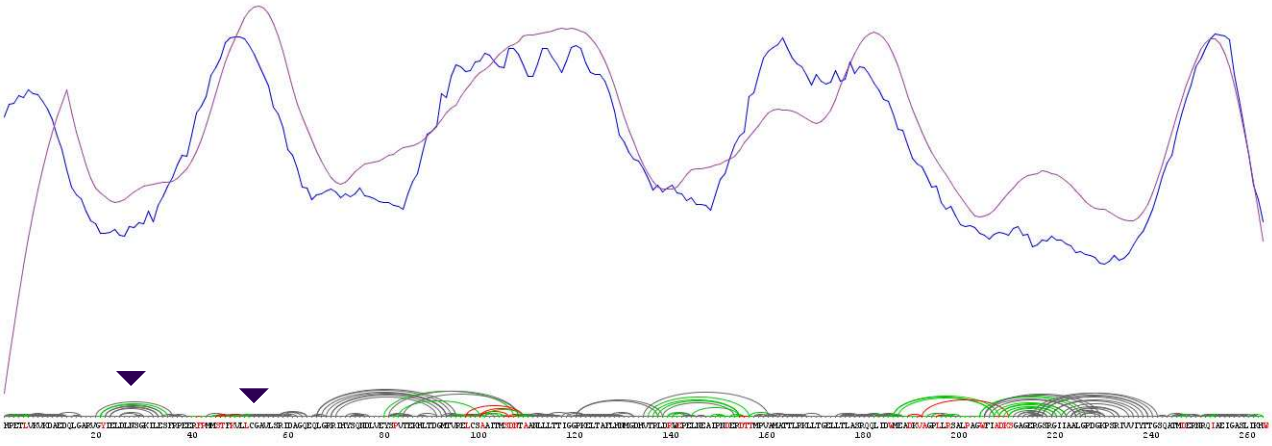
A second issue is the level at which SCHEMA takes interactions into account. SCHEMA is a window based approach and operates therefore only on connections where the participating residues have a maximal distance of 30 (a window of 15 residues to both sides). Figure 4.8 b shows the connections from which the SCHEMA is calculated. The first triangle in Figure 4.8 b points out a position where 7 connections form a, so called, hair pin formation. SCHEMA has a low score here because only 7 connections are present and, due to the structure they form, only the center connection is in the area which is emphasized by the implicit weighting of the nested sums. The position pointed out by the second triangle has, in contrast, a great number of interactions with residue distance of 3 or 4. This highly connected area forms a α helix and has a high SCHEMA score, although Voigt *et al* report that it appears to be beneficial to cut in the middle of a helix [1]. It remains to be demonstrated which low level structure is more important to be kept intact or if it is even more beneficial to take longer ranging connections into account as well.

Thus, future investigations will concentrate on the influence of long range interactions which are currently not involved in the recombination site determination. Figure 4.8 c shows all the contacts the tertiary structure consists of. It is clear that connections spanning more than half of the protein can not be maintained. The idea behind recombination was not to retain the exact same set of connections because the intention was to exploit a new structure. Recombining sequence parts from two similar proteins promises to keep the long range interactions intact because the parental sequences are likely to have residues with similar properties at similar positions. Even though long range connections are broken by the cut they can be rebuilt with similar residues from the sequence part from the other parent after the recombination.

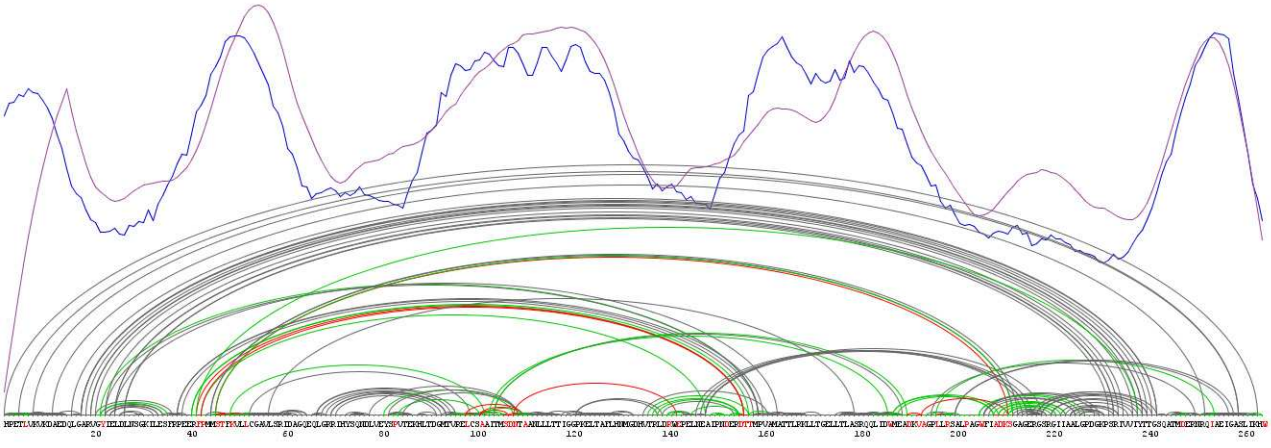
Therefore it might be a good guideline to look at the functional domains of the protein that are already known (see Table 4.14 and Figure 4.9). It seems to be beneficial to cut inside a domain. Beneficial changes in functional domains are more likely to yield a performance improvement than chances outside the active area. Recombining the first half of the domain from parent A with the second half of the domain from parent B combines two parts that have been independently changed by evolution. The parent proteins have a similar function; therefore the overall scaffold is likely to be the same while connections with a smaller range may differ significantly. Combining those characteristics of local connections from different parents may result in a performance improvement due to, say, a longer half-life or higher thermo stability.



(a)



(b)



(c)

Fig. 4.8: β-lactamase sequence (red letters are the critical residues) along with the predicted SCHEMA profile (blue) and the calculated profile (lavender). Contacts between residues are indicated as arcs where green is an interaction between a normal residue and a critical one, red is an interaction between two critical residues and grey between two normal residues. (a) critical residues with the contacts they are involved in (b) only contacts are displayed where the distance between the two contacting residues is less then 30. This is the level the SCHEMA calculation takes into account. (c) full contact visualization

ProDom domains producing High-scoring Segment Pairs					
Position	ProDom domain	Score	E value	Idendity	Length
142-281	#PD111549	611	3^{-64}	98%	138
30-132	#PD000503	508	2^{-52}	98%	103
1-29	#PD332083	153	4^{-11}	100%	29
141-177	#PD861210	120	2^{-07}	62%	37
45-141	#PD858658	89	0.001	27%	133

Tab. 4.14: Domains in Q4AE67 [75]

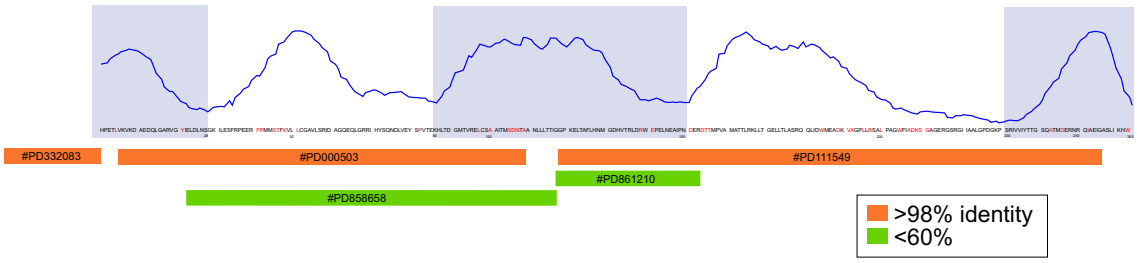


Fig. 4.9: Domains in Q4AE67. The connections and calculated SCHEMA score are on the basis of the β-lactamase protein in PDB 1ZG4 (because the tertiary structure was required). The sequence for 1ZG4 lacked the first 23 amino acids; therefore the first domain is listed without the amino acid sequence. Alignment according to Myers and Miller, CABIOS (1989) 4:11-17

Future approaches will therefore be concerned with two issues the current SCHEMA score is not able to handle. Firstly, operating on the whole protein instead of the window based approach which yields the benefit of taking long range dependencies into account. Secondly, including the information of the actual protein sequences that should be combined instead of examining both proteins independently.

5. CONCLUSION

The work has presented and evaluated an approach to predict the SCHEMA score from the protein sequence. The SCHEMA score has proven to be biologically plausible and was used in several protein design attempts [1]. To find the right means, two different but previously successfully utilized machine learning techniques were surveyed: Neural Networks and Support Vector Regression (SVR).

The models were trained on a large set of protein data to predict the structural disruption score as determined by the original SCHEMA algorithm. The machine learning methods were presented with a window of sequence residues as input. As expected, the prediction on the plain sequence was not successful as the contact map is not easy to predict. Just as in the tertiary structure prediction, an intermediate step over the predicted *secondary structure* made the SCHEMA score prediction possible. The secondary structure was predicted by a Neural Network method, the Continuum Secondary Structure Predictor introduced by Bodén *et al.* [15]. The best method among the machine learning approaches presented with the predicted 3- and 8-state *secondary structure* was the bidirectional recurrent neural network (BRNN) with a correlation coefficient of 0.88.

Only little differences were observed between the presentation of 3-state or 8-state secondary structure. The good performance of the single layer neural network and the SVR using the linear Kernel strongly indicates a linear dependency between the *secondary structure* and the SCHEMA score. The window-based SCHEMA calculation takes only connections into consideration which span maximally 30 residues. This is roughly the area that determines the *secondary structure* element of this sequence part. It is therefore not surprising that the SCHEMA score prediction is supported by the *secondary structure*. This assumption is promoted by the observation that providing *contact numbers* as additional input is not resulting in better performance. Since the SCHEMA calculation is window-based and the *secondary structure* already covers information about these local connections, the only additional information the *contact numbers* could provide are long range interactions. Connections exceeding the window size of the calculation are not taken into account anyway and are therefore not aiding in the prediction.

Besides the *contact numbers*, the *solvent accessibility* score was trialed as well as an additional

input feature to the *secondary structure*. Since the prediction accuracy decreased, it is likely that presenting the information about solvent accessibility coded as a score is not the right means and provides therefore confusing information to the predictor.

However, the correlation coefficient of 0.88 strongly supports the hypothesis that the SCHEMA score is predictable from sequence although an intermediate step - predicting *secondary structure* - is necessary. This machine learning method enables biologists to freely choose candidate proteins on the basis of functional properties, and not be limited to those for which full structural information is available. The presented model represents the first predictor of a structural disruption score and should be of considerable benefit for protein design efforts because it provides access to the full power of *in silico* protein design.

In future research, the predicted SCHEMA score can function as one component in a larger predictor complex taking other information into account to predict recombination sites more accurately. Future attempts should therefore focus on the influence of long range interactions and ideally take the whole connection set into account. The original SCHEMA calculation suffers form the same inabilities which makes it necessary to move on to an approach, that not only takes long range dependencies into account but also addresses the issue of recombining the determined blocks. This is supported by resent results from Arnold *and coworkers* [76].

The SCHEMA score could therefore function as a heuristic to identify the area of possible recombination sites. The exact position can then be exploited by taking the parental sequences into account. The sequence part which is to be combined can be searched for residues which are capable of functioning as connection partners for connections which are exceeding the cutting site and are otherwise considered as broken [76].

BIBLIOGRAPHY

[1] C. A. Voigt, C. Martinez, Z.-G. Wang, S. L. Mayo, and F. H. Arnold, "Protein building blocks preserved by recombination," *Nat Struct Biol*, vol. 9, no. 7, pp. 553–558, Jul 2002.

[2] B. I. Dahiyat and S. L. Mayo, "De novo protein design: fully automated sequence selection," *Science*, vol. 278, no. 5335, pp. 82–87, Oct 1997.

[3] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri, "Exploiting the past and the future in protein secondary structure prediction," *Bioinformatics*, vol. 15, no. 11, pp. 937–946, Nov 1999.

[4] M. Ostermeier, J. H. Shim, and S. J. Benkovic, "A combinatorial approach to hybrid enzymes independent of DNA homology," *Nat Biotechnol*, vol. 17, no. 12, pp. 1205–1209, Dec 1999.

[5] S. Lutz, M. Ostermeier, G. L. Moore, C. D. Maranas, and S. J. Benkovic, "Creating multiple-crossover DNA libraries independent of sequence identity," *Proc Natl Acad Sci U S A*, vol. 98, no. 20, pp. 11 248–11 253, Sep 2001.

[6] V. Sieber, C. A. Martinez, and F. H. Arnold, "Libraries of hybrid proteins from distantly related sequences," *Nat Biotechnol*, vol. 19, no. 5, pp. 456–460, May 2001.

[7] W. P. Stemmer, "Rapid evolution of a protein in vitro by DNA shuffling," *Nature*, vol. 370, no. 6488, pp. 389–391, Aug 1994.

[8] K. Hiraga and F. H. Arnold, "General method for sequence-independent site-directed chimeragenesis," *J Mol Biol*, vol. 330, no. 2, pp. 287–296, Jul 2003.

[9] M. M. Meyer, J. J. Silberg, C. A. Voigt, J. B. Endelman, S. L. Mayo, Z.-G. Wang, and F. H. Arnold, "Library analysis of SCHEMA-guided protein recombination," *Protein Sci*, vol. 12, no. 8, pp. 1686–1693, Aug 2003.

[10] Y. Yia and M. Levitt, "Roles of mutation and redcombination in the evolution of protein thermodynamics," *PNAS*, vol. 99, no. 16, pp. 10 382–10 387, 2002.

[11] J. Cheng, A. Randall, M. Sweredoski, and P. Baldi, "SCRATCH: a Protein Structure and Structural Feature Prediction Server," *Nucleic Acids Research*, vol. Special Issue on Web servers, p. in press, 2005.

[12] P. Baldi and G. Pollastri, "Machine learning structural and functional proteomics," *IEEE Intelligent Systems. Special Issue on Intelligent Systems in Biology*, vol. 17, no. 2, pp. 28–35, March/April 2002. [Online]. Available: http://www.ics.uci.edu/ pfbaldi/publications/journals/ieeev2.pdf

[13] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *Journal of Molecular Biology*, vol. 292, pp. 195–202, 1999.

[14] S. Hua and Z. Sun, "A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach,," *Journal of Molecular Biology*, vol. 308, no. 2, pp. 397–407, 2001.

[15] M. Bodén, Z. Yuan, and T. L. Bailey, "Prediction of protein continuum secondary structure with probabilistic models," *Submitted*, 2005.

[16] G. Pollastri, P. Baldi, P. Fariselli, and R. Casadio, "Prediction of coordination number and relative solvent accessibility in proteins," *Proteins: Structure, Function, and Genetics*, vol. 47, pp. 142–153, 2002.

[17] Z. Yuan, "Better prediction of protein contact numbers with support vector regression," *Submitted*, 2005.

[18] B. Rost, "Twilight zone of protein sequence alignments," *Protein Eng*, vol. 12, no. 2, pp. 85–94, Feb 1999.

[19] ——, "Review: protein secondary structure prediction continues to rise," *J Struct Biol*, vol. 134, no. 2-3, pp. 204–218, May 2001.

[20] ——, "PHD: predicting one-dimensional protein structure by profile-based neural networks," *Methods Enzymol*, vol. 266, pp. 525–539, 1996.

[21] K. T. Simons, R. Bonneau, I. Ruczinski, and D. Baker, "Ab initio protein structure prediction of CASP III targets using ROSETTA," *Proteins*, vol. Suppl 3, pp. 171–176, 1999.

[22] C. Bystroff, V. Thorsson, and D. Baker, "HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins," *J Mol Biol*, vol. 301, no. 1, pp. 173–190, Aug 2000.

[23] C. Bystroff and D. Baker, "Blind predictions of local protein structure in CASP2 targets using the I-sites library," *Proteins*, vol. Suppl 1, pp. 167–171, 1997.

[24] X. Yuan and C. Bystroff, "Non-sequential structure-based alignments reveal topology-independent core packing arrangements in proteins," *Bioinformatics*, vol. 21, no. 7, pp. 1010–1019, Apr 2005.

[25] A. Kinjo and K. Nishikawa, "Recoverable one-dimensional encoding of protein three-dimensional structures," *Bioinformatics*, Feb 2005.

[26] M. H. Hecht, A. Das, A. Go, L. H. Bradley, and Y. Wei, "De novo proteins from designed combinatorial libraries," *Protein Sci*, vol. 13, no. 7, pp. 1711–1723, Jul 2004.

[27] J. R. Desjarlais and N. D. Clarke, "Computer search algorithms in protein modification and design," *Curr Opin Struct Biol*, vol. 8, no. 4, pp. 471–475, Aug 1998.

[28] C. A. Voigt, D. B. Gordon, and S. L. Mayo, "Trading accuracy for speed: A quantitative comparison of search algorithms in protein sequence design," *J Mol Biol*, vol. 299, no. 3, pp. 789–803, Jun 2000.

[29] D. T. Jones, "De novo protein design using pairwise potentials and a genetic algorithm," *Protein Sci*, vol. 3, no. 4, pp. 567–574, Apr 1994.

[30] L. Wernisch, S. Hery, and S. J. Wodak, "Automatic protein design with all atom force-fields by exact and heuristic optimization," *J Mol Biol*, vol. 301, no. 3, pp. 713–736, Aug 2000.

[31] J. Desmet, M. De Maeyer, and I. Lasters, "The dead-end elimination theorem and its use in protein side-chain positioning," *Nature*, vol. 356, no. 5, pp. 539–542, Apr 1992.

[32] F. Arnold, *Evolutionary Protein Design*. New York: Academic Press, 2001.

[33] M. Go, "Correlation of DNA exonic regions with protein structural units in haemoglobin," *Nature*, vol. 291, no. 5810, pp. 90–92, May 1981.

[34] ——, "Modular structural units, exons, and function in chicken lysozyme," *Proc Natl Acad Sci U S A*, vol. 80, no. 7, pp. 1964–1968, Apr 1983.

[35] S. J. de Souza, M. Long, L. Schoenbach, S. W. Roy, and W. Gilbert, "Intron positions correlate with module boundaries in ancient proteins," *Proc Natl Acad Sci U S A*, vol. 93, no. 25, pp. 14 632–14 636, Dec 1996.

[36] W. Gilbert, S. J. de Souza, and M. Long, "Origin of genes," *Proc Natl Acad Sci U S A*, vol. 94, no. 15, pp. 7698–7703, Jul 1997.

[37] A. Crameri, S. A. Raillard, E. Bermudez, and W. P. Stemmer, "DNA shuffling of a family of genes from diverse species accelerates directed evolution," *Nature*, vol. 391, no. 6664, pp. 288–291, Jan 1998.

[38] J. M. Bacher, B. D. Reiss, and A. D. Ellington, "Anticipatory evolution and DNA shuffling," *Genome Biol*, vol. 3, no. 8, p. REVIEWS1021, Jul 2002.

[39] "Swiss prot current holdings," 29 Mar 2005. [Online]. Available: http://au.expasy.org/sprot/relnotes/relstat.html

[40] "Pdb current holdings," 29 Mar 2005. [Online]. Available: http://www.rcsb.org/pdb/holdings.html

[41] F. Rosenblatt, "The perceptron: a probabilistic model for information storage in the brain," *Psychological Review*, vol. 65, pp. 386–408, 1958.

[42] V. Vapnik, *Statistical learning theory.* Wiley, 1998.

[43] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577–2637, 1983.

[44] B. Rost and C. Sander, "Third generation prediction of secondary structures," *Methods Mol Biol*, vol. 143, pp. 71–95, 2000.

[45] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *J Mol Biol*, vol. 292, no. 2, pp. 195–202, Sep 1999.

[46] I. Y. Y. Koh, V. A. Eyrich, M. A. Marti-Renom, D. Przybylski, M. S. Madhusudhan, N. Eswar, O. Grana, F. Pazos, A. Valencia, A. Sali, and B. Rost, "EVA: Evaluation of protein structure prediction servers." *Nucleic Acids Res*, vol. 31, no. 13, pp. 3311–3315, 2003.

[47] M. Eisenstein, "Problem solved?" *Nature Methods*, vol. 2, pp. 162–194, 2005.

[48] M. Kuhn, J. Meiler, and D. Baker, "Strand-loop-strand motifs: prediction of hairpins and diverging turns in proteins," *Proteins*, vol. 54, no. 2, pp. 282–288, Feb 2004.

[49] A. Cleeremans, *Mechanisms of implicit learning.* Cambridge, MA: MIT Press, 1993.

[50] C. A. F. Andersen, A. G. Palmer, S. Brunak, and B. Rost, "Continuum secondary structure captures protein flexibility," *Structure*, vol. 10, pp. 175–184, 2002.

[51] M. N. Nguyen and J. C. Rajapakse, "Prediction of protein relative solvent accessibility with a two-stage SVM approach," *Proteins*, vol. 59, no. 1, pp. 30–37, Apr 2005.

[52] M. J. Thompson and R. A. Goldstein, "Predicting solvent accessibility: higher accuracy using Bayesian statistics and optimized residue substitution classes," *Proteins*, vol. 25, no. 1, pp. 38–47, May 1996.

[53] S. Pascarella, R. De Persio, F. Bossa, and P. Argos, "Easy method to predict solvent accessibility from multiple protein sequence alignments," *Proteins*, vol. 32, no. 2, pp. 190–199, Aug 1998.

[54] X. Li and X. M. Pan, "New method for accurate prediction of solvent accessibility from protein sequence," *Proteins*, vol. 42, no. 1, pp. 1–5, Jan 2001.

[55] G. Pollastri, P. Baldi, P. Fariselli, and R. Casadio, "Prediction of coordination number and relative solvent accessibility in proteins," *Proteins*, vol. 47, no. 2, pp. 142–153, May 2002.

[56] S. Ahmad and M. M. Gromiha, "NETASA: neural network based prediction of solvent accessibility," *Bioinformatics*, vol. 18, no. 6, pp. 819–824, Jun 2002.

[57] H. Naderi-Manesh, M. Sadeghi, S. Arab, and A. A. Moosavi Movahedi, "Prediction of protein surface accessibility with information theory," *Proteins*, vol. 42, no. 4, pp. 452–459, Mar 2001.

[58] G. Gianese, F. Bossa, and S. Pascarella, "Improvement in prediction of solvent accessibility by probability profiles," *Protein Eng*, vol. 16, no. 12, pp. 987–992, Dec 2003.

[59] R. Adamczak, A. Porollo, and J. Meller, "Accurate prediction of solvent accessibility using neural networks-based regression," *Proteins*, vol. 56, no. 4, pp. 753–767, Sep 2004.

[60] B. Rost and C. Sander, "Conservation and prediction of solvent accessibility in protein families," *Proteins*, vol. 20, no. 3, pp. 216–226, Nov 1994.

[61] H. Kim and H. Park, "Prediction of protein relative solvent accessibility with support vector machines and long-range interaction 3D local descriptor," *Proteins*, vol. 54, no. 3, pp. 557–562, Feb 2004.

[62] K. Nishikawa and T. Ooi, "Prediction of the surface-interior diagram of globular proteins by an empirical method," *Int J Pept Protein Res*, vol. 16, no. 1, pp. 19–32, Jul 1980.

[63] A. Kabakcioglu, I. Kanter, M. Vendruscolo, and E. Domany, "Statistical properties of contact vectors," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 65, no. 4 Pt 1: 041904, Apr 2002.

[64] U. Hobohm, M. Scharf, R. Schneider, and C. Sander, "Selection of representative protein data sets," *Protein Science*, vol. 1, pp. 409–417, 1992.

[65] B. Thiruv, G. Quon, S. A. Saldanha, and B. Steipe, "Nh3D: a reference dataset of non-homologous protein structures," *BMC Struct Biol*, vol. 5, p. 12, Jul 2005.

[66] O. Carugo, "Predicting residue solvent accessibility from protein sequence by considering the sequence environment," *Protein Eng*, vol. 13, no. 9, pp. 607–609, Sep 2000.

[67] J. J. Ward, L. J. McGuffin, B. F. Buxton, and D. T. Jones, "Secondary structure prediction with support vector machines," *Bioinformatics*, vol. 19, no. 13, pp. 1650–1655, 2003.

[68] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri, "Exploiting the past and the future in protein secondary structure prediction," *Bioinformatics*, vol. 15, pp. 937–946, 1999.

[69] S. Sundararajan and S. S. Keerthi, "Predictive approaches for choosing hyperparameters in gaussian processes," *Neural Computation*, vol. 13, no. 5, pp. 1103–1118, 2001.

[70] C. C. Chang and C. J. Lin, "LIBSVM 2.0: Solving different support vector formulations." [Online]. Available: http://www.csie,ntu.edu.tw/˜ cjlin/libsvm

[71] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999. [Online]. Available: citeseer.ist.psu.edu/opitz99popular.html

[72] H. Saigo, J.-P. Vert, N. Ueda, and T. Akutsu, "Protein homology detection using string alignment kernels," *Bioinformatics*, vol. 20, no. 11, pp. 1682–1689, Jul 2004, evaluation Studies.

[73] W. Huang, J. Petrosino, M. Hirsch, P. S. Shenkin, and T. Palzkill, "Amino acid sequence determinants of beta-lactamase structure and activity," *J Mol Biol*, vol. 258, no. 4, pp. 688–703, May 1996.

[74] A. Dubey, M. J. Realff, J. H. Lee, and A. S. Bommarius, "Support vector machines for learning to identify the critical positions of a protein," *J Theor Biol*, vol. 234, no. 3, pp. 351–361, Jun 2005.

[75] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Z. Zheng Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic Acids Res*, vol. 25, pp. 3389–3402, 1997.

[76] C. R. Otey, M. Landwehr, J. B. Endelman, K. Hiragam, J. D. Bloom, and F. H. Arnold, "Site-directed recombination creates an artificial family of cytochromes P450," *PNAS*, vol. in press, 2005.