Predicting structural disruption of proteins caused by crossover

Denis C. Bauer*, Mikael Bodén*§, Ricarda Thier†, Zheng Yuan‡
*School of Information Technology and Electrical Engineering
†School of Biomedical Sciences
‡Institute for Molecular Bioscience and ARC Centre in Bioinformatics
The University of Queensland
QLD 4072
Australia
§Corresponding author. mikael@itee.uq.edu.au

***Abstract*— We present a machine learning model that predicts a structural disruption score from a protein's primary structure. SCHEMA was introduced by Frances Arnold and colleagues as a method for determining putative recombination sites of a protein on the basis of the full (PDB) description of its structure. The present method provides an alternative to SCHEMA that is able to determine the same score from sequence data only. Circumventing the need for resolving the full structure enables the exploration of yet unresolved and even hypothetical sequences for protein design efforts. Deriving the SCHEMA score from a primary structure is achieved using a two step approach: first predicting a secondary structure from the sequence and then predicting the SCHEMA score from the predicted secondary structure. The correlation coefficient for the prediction is 0.88 and indicates the feasibility of replacing SCHEMA with little loss of precision.**

I. Introduction

This work examines the ability and accuracy of machine learning approaches in predicting a structural disruption score from amino acid sequence data alone. The SCHEMA-score was introduced by Voigt *et al.* to quantify the structural disruption caused by using a sequence position as a crossover site in recombination-based protein design [1].

Recombination has proven to be very powerful in protein design [2], [3], [4]. Unlike traditional protein design methods where new proteins are built from scratch [5], recombination deals with native sequence parts whose properties are already known. Recombination principles systematically reduce the combinatorial space of possible sequences to a practically explorable area.

A pre-requisite for effective directed recombination is the knowledge of possible recombination sites. The assumption behind SCHEMA is that function is preserved by retaining structural components. SCHEMA assists in dividing a protein into segments that are likely to fold independently from the rest of the protein. The segments form building blocks sampled from a number of different parent proteins and can be recombined with the main structural features left intact [1].

Essentially Voigt *et al.* estimate preserved function from preserved contact between residues. Two residues are defined to be in contact if they are within a predefined distance (4.5 Å) of each other. SCHEMA calculates for each residue the number of connections that break if the recombination is set at this position. The minima of the calculated profile over the sequence indicate possible recombination sites [1].

Importantly, the recombination sites identified by SCHEMA favourably match successful recombination sites as established by *in vitro* experiments [1]. Furthermore, protein design experiments with SCHEMA guided recombination have led to highly functional protein hybrids [1], [4].

The SCHEMA calculation, however, requires the full tertiary description of the protein (as in the Protein Data Bank). This requirement severely limits the number of candidate proteins to the small group that are already resolved by X-ray crystallography or NMR. Due to the expensive, time-consuming 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.

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 prediction is disengaged from the tertiary structure, we can fully tap the power of *in silico* protein design.

The present work presents and evaluates an approach to predict the SCHEMA score from the protein sequence. To find the right means, two different but previously successfully utilised machine learning techniques are surveyed: Neural Networks and Support Vector Regression (SVR). The models are trained on a large set of protein data to predict the structural disruption score as determined by the original SCHEMA algorithm. Each point in the sequence is processed by using a window of sequence residues as input. We evaluate models that are only presented with plain sequence data, models that are presented with a predicted secondary structure, and models that are presented with predicted residue contact numbers combined with secondary structure.

II. Material and Methods

A. Data set

We use a data set consisting of 945 proteins taken from the Protein Data Bank. The data set represents a diverse range of proteins and has no pairs with more than 25% sequence similarity [6].

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

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

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. Contact is defined as two residues within the spatial distance of 4.5 Å.

The way the sums are nested creates an implicit weighting. 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.

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 used by Voigt *et al.* The complete binary contact map was derived for each of the proteins in the dataset by employing a Euclidian cut-off distance of 4.5 Å. Additionally, the residues themselves and immediate neighbours are defined to be in contact.

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. Thus, we report on models that use a zero-one bounded transformation:

$$S' = tanh\left(\frac{S}{n}\right) \tag{2}$$

where n is a normalization constant $\left(n = \frac{MAX(S_i)}{2} = 874\right)$.

2) The input sequence data: The study evaluates three non-exclusive means for presenting the 945-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. 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 [7], [8].

The *secondary structure* of each residue – either three-state (ss3) or eight-state (ss8) [9] – can be predicted from the sequence data. We use the Continuum Secondary Structure Predictor [10] which produces a probability for each secondary structure state (both for three-state and eight-state). The secondary structure model is based on a recently proposed scheme by Andersen *et al.* [11] 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 [10]. The classification accuracy achieved by thresholding this probabilistic predictor is, at $Q_3 = 77.3$, on par with standard categorical secondary structure predictors.

We take the predicted probabilities of the continuum secondary structure as representing 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.

We also experiment with another structural feature – the residue *contact number* (co) – which describes the number of contacts the residue has with other residues in the same protein. The contact number is a single value for each residue.

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. [12]. We used the SVR Contact Number Predictor [13] which predicts the contact number for each residue from the primary structure with an correlation coefficient of 0.70. Contact numbers are normalized by the following equation:

$$Co' = \frac{Co - \langle Co \rangle}{\sqrt{(Co - \langle Co \rangle)^2}} \tag{3}$$

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

B. Predictors

We evaluate two major types of machine learning algorithms, 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 [7], [14], [10], contact number and solvent accessibility prediction [12], [13]). To make the comparison straight forward we also configure models in accordance with previous studies of predicting structural features. The window size of all models is set to 15 residues: the residue for which the SCHEMA score is predicted and then 7+7 residues immediately upstream and downstream, respectively. This configuration has been found to be close to optimal for most contact number and secondary structure predictors. Algorithm-specific parameter values are provided in the following.

TABLE I

KERNEL

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

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. The network is standardly trained using gradient descent to minimise the error as measured on the single output node. The learning rate is $\eta = 0.001$. A variety of hidden node numbers h (including not using a hidden layer at all) are trialled. A sigmoidal output function is used.

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 [15], [16]. 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” [17]. 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 are changed. A total of 40,000 sequences were presented in random order before we stopped training. In preliminary studies this number was seen as sufficient for convergence.

2) Support Vector Regression: Recent findings suggest that Support Vector Regression (SVR) exceeds the accuracy reached by many neural networks [18], [13]. Essentially, support vector regression operates by finding so-called support vectors that collectively represent the function in a feature space. Support vector regression minimizes the ϵ-tube which is wrapped around the approximated function to cover most of the data points. Importantly, a kernel function maps the input sequence encoding into the feature space. We examine optimisation using ϵ-SVR and ν-SVR with the same protein data set as used for the neural networks. The standard stopping criterion is 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 I, with $\gamma = 1, c = 1$ and $d = 3$). We use the LIBSVM implementation of the optimisation procedures [19].

C. Testing

Preliminary simulations with the used models showed that the differences between 10-fold and 2-fold crossvalidation are negligible (not shown). To minimize the computational time required, we consistently use 2-fold crossvalidation to develop and test models.

1) Performance Measure: Two performance measures are 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.

$$r = \frac{\langle (t_i - \langle t_i \rangle) \cdot (p_i - \langle p_i \rangle) \rangle}{\sqrt{\langle (t_i - \langle t_i \rangle) \rangle} \cdot \sqrt{\langle (p_i - \langle p_i \rangle) \rangle}} \qquad (4)$$

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.

$$DevA = \frac{\sqrt{\langle (p_i - t_i)^2 \rangle}}{\sqrt{\langle (t_i - \langle t_i \rangle)^2 \rangle}} \qquad (5)$$

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).

III. RESULTS

A. 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 II and Table III).

B. The secondary structure as input

The results when secondary structure is used as input for the two neural network architectures are shown in Table II and Table III. The results when secondary structure is used as input for the support vector regression algorithm are shown in Table IV and Table V.

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). The predicted outputs of the BRNN for three sequences are shown in Figure 1.

TABLE II

THE PERFORMANCE OF FFNN (USING 2-FOLD CROSSVALIDATION, A SECONDARY STRUCTURE 15-RESIDUE INPUT WINDOW, AND TRAINING FOR 40,000 SEQUENCES).

FFNN			
Inputset	h	r	devA
ps	0	0.56	0.95
	20	0.56	0.95
	40	0.56	0.95
ss3	0	0.86	0.57
	20	0.86	0.57
	40	0.86	0.57
ss8	0	0.86	0.56
	20	0.85	0.69
	40	0.85	0.58

TABLE III

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

BRNN			
Inputset	h	r	devA
ps	7+7	0.66	0.90
ss3	7+7	0.88	0.52
ss8	7+7	0.87	0.55

TABLE IV

The performance of ϵ-SVR (using 2-fold crossvalidation and secondary structure as input).

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

TABLE V

The performance of ν-SVR (using 2-fold crossvalidation and secondary structure as input).

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

TABLE VI

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).

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

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. 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.

C. 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.

As shown in Table VI, Table VII and Table VIII, 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 a possible inaccuracy of the contact number predictor. We carried out several tests with the FFNN trained on secondary structure and contact number as determined directly from the contact map derived from the PDB descriptions. The difference in using this observed contact number and the predicted contact number is negligible (Table VI), which supports the conclusion that contact numbers are not aiding in improving the accuracy beyond the contribution of the secondary structure.

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

The impact of using alternative structural features, e.g. predicted solvent accessibility, remains to be investigated.

TABLE VII

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).

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

TABLE VIII

The performance of ϵ-SVR using the secondary structure and predicted contact number as input (2-fold crossvalidation and secondary structure with contact numbers as input).

ϵ-SVR			
Inputset	kernel	r	devA
ss3 co	linear	0.82	0.63
	hbf	0.80	0.66
ss8 co	linear	0.85	0.60
	hbf	0.71	0.84

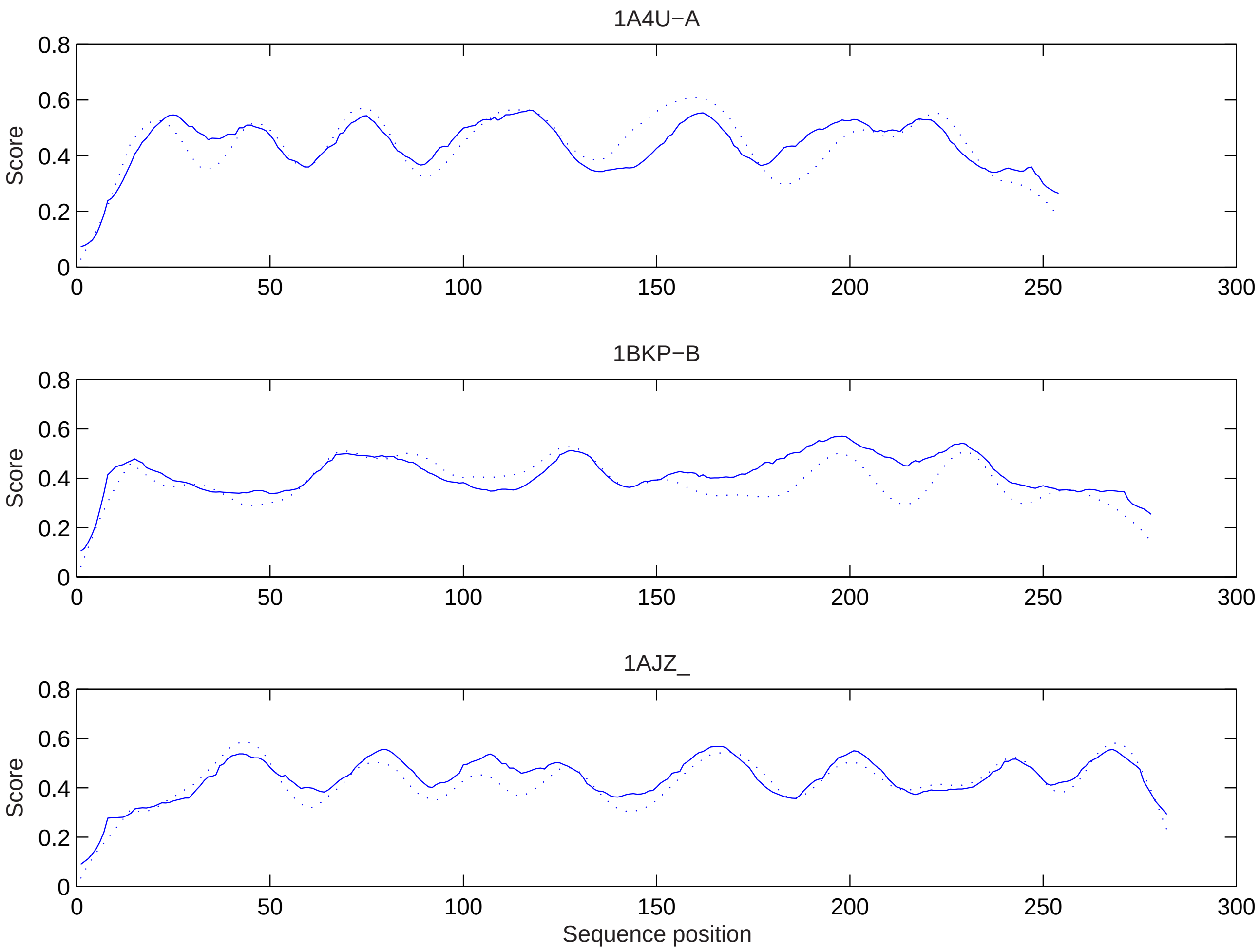

Fig. 1. Examples of predictions produced by the BRNN with ss3 input ($h = 7 + 7$, $w = 15$, $r = 0.88$). The solid line represents the actual SCHEMA score and the dashed line represents the predicted score.

D. *Evaluation of the predicted minima*

The most valuable information within the SCHEMA score are according to Voigt *et al.* the minima. It is therefore not imperative to approximate the target function T exactly as long as the minima are the same. The distance between the positions of the minima in the predicted function P 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 is smoothed with a linear kernel: the mean of the values within a window. The window size for the target function is $w = 3$. For the predicted function the window size is 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 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.

$$C_{i,j} = Min \begin{cases} C_{i-1,j-1} + abs(P_j - T_i), & \text{if } \langle T_i, P_j \rangle; \\ C_{i,j-1} + 10, & \text{if } \langle -, P_j \rangle; \\ C_{i-1,j} + 10, & \text{if } \langle T_i, - \rangle. \end{cases} \tag{6}$$

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

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 data set is shown in in Figure 2.

IV. Conclusion

The goal was to develop a machine learning approach that is able to predict the SCHEMA score – a very successful heuris-

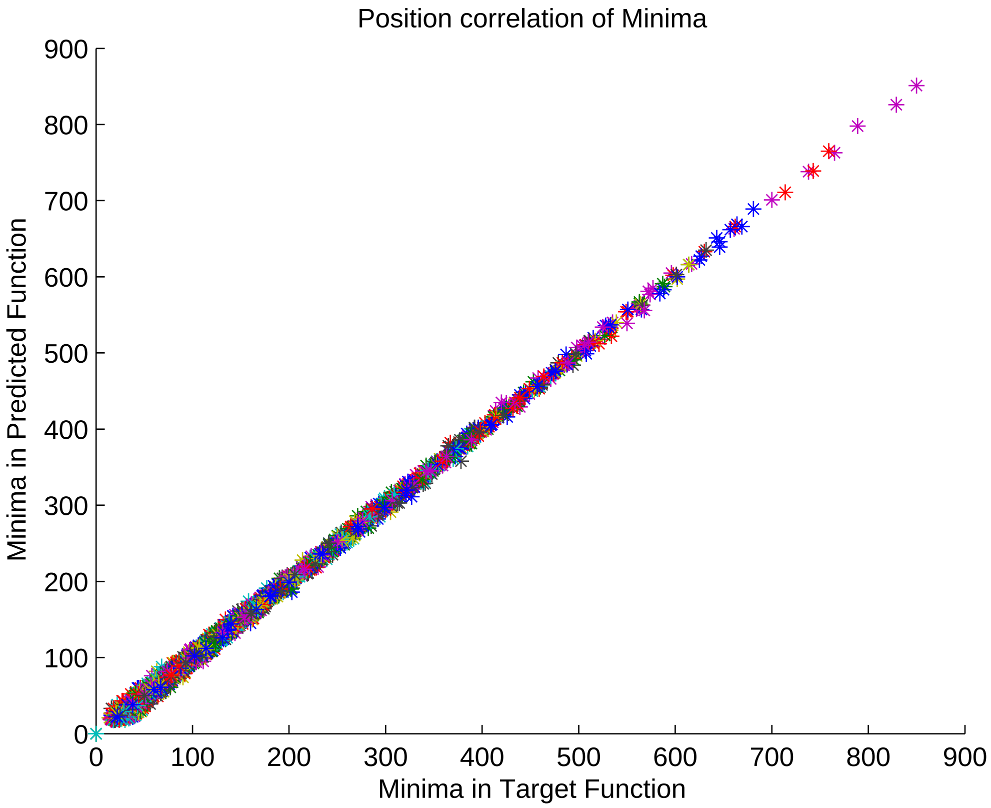

Fig. 2. A scatter plot of the position of minima m_i in the predicted score against the position in the target function for all minima M in the data set. Ideally, the plot forms a perfect diagonal.

tic in the exploration of protein design. Prediction from the primary structure enables an *in silico* study of all proteins and not only the ones where the complete structural information is already known.

To the best of our knowledge, the presented model represents the first predictor of a structural disruption score and should be of considerable benefit for protein design efforts. The prime model is a cascaded model that produces an accurate SCHEMA score ($r = 0.88$) from the sequence data alone.

Models trained directly on the primary structure of the proteins have severe difficulties in finding a reasonable generalisation. Predicting from secondary structure, however, leads to successful results. It seems that when additional structural features like secondary structure and contact number are used the SCHEMA score can successfully be predicted. Fortunately, such features are themselves predictable from the primary structure. We show how a secondary structure predictor is used to provide input data for the SCHEMA score predictor. The best predictor in this study is the bidirectional recurrent neural network.

References

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

[3] 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.

[4] 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.

[5] 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.

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

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

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.

[13] Z. Yuan, “Better prediction of protein contact numbers with support vector regression,” *Submitted*, 2005.

[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] 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.

[16] 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.

[17] 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.

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

[19] 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