Technical University of Denmark

Prediction of protein structural features by use of artificial neural networks

Petersen, Bent; Lundegaard, Claus; Petersen, Thomas Nordahl

Publication date:
2011

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Petersen, B., Lundegaard, C., & Petersen, T. N. (2011). Prediction of protein structural features by use of artificial neural networks. Kgs. Lyngby, Denmark: Technical University of Denmark (DTU).

Prediction of protein structural features by use of artificial neural networks

Bent Petersen

31 March, 2011

CENTERFO
RBIOLOGI
CALSEQU
ENCEANA
LYSIS CBS

Preface

This thesis was prepared at the Center for Biological Sequence Analysis (CBS), Department of Systems Biology, at the Technical University of Denmark (DTU) as the requirement for acquiring the Ph.D. degree. The Ph.D. was founded by a DTU scholarship, and the NetSurfP project was further founded by the Center for Disease Systems Biology under a grant from the Villum Kann Rasmussen foundation http://www.vkr-fondene.dk, and the EU Commission, BioSapiens (NoE), FP6, contract no.: LSHG-CT-2003-503265.

Almost all the work was carried out at the Center for Biological Sequence Analysis under the supervision of Associate Professor Claus Lundegaard and Associate Professor Thomas Nordahl Petersen. The work presented in chapter 5 was done in collaboration with Martin Willemoës at the Department of Biology, University of Copenhagen (KU) in 2010.

Lyngby, March 2011

Bent Petersen

Contents

Abstract

In the past decades we have seen an exponential growth of biological sequence data. The cost for DNA sequencing has dropped significantly since the announcement of the first sequenced genome and newly sequenced genomes are published almost every week. Publicly available genetic sequence databases like for example GenBank are increasing considerably in size and GenBank currently contains more than 132 million sequences. Similar the Protein Data Bank currently contains more than 71,000 experimentally determined structures of nucleic acids, proteins and nucleic acid/protein complexes. There is a huge over-representation of DNA sequences when comparing the amount of experimentally verified proteins with the amount of DNA sequences. The academic and industrial research community therefore has to rely on structure predictions instead of waiting for the time consuming experimentally determined structure data.

This thesis describes the development of two new tools to study such genetic sequence data. NetSurfP was developed to predict the surface accessibility of amino acids in amino acid sequences. Knowledge of the degree of surface exposure of an amino acid is valuable and has been used to enhance the understanding of a variety of biological problems, including protein-protein interaction, prediction of epitopes and active sites. Following NetSurfP, NetTurnp was developed for the prediction of β-turn occurrence. Using secondary structure and surface accessibility predictions from NetSurfP, a better understanding and improvement of the performance for the prediction of β-turns was obtained. β-turns are very interesting in the way that they are the most abundant type of turn structures, and approximately 25% of all amino acids in protein structures are located in a β-turn.

In bioinformatics speed and accuracy is an important factor, hence the developed tools are expected to return a result in a rapid and efficient manner. Our way of solving that problem was to pre calculate protein sequence data. Currently, more than 500,000 protein sequences are in the local cache.

In relation to surface exposure, a third project dealt with the prediction of discontinuous B-cell epitopes. Here Half Sphere Exposure (HSE) was integrated in an existing prediction method. HSE is a measure of solvent exposure where the upper and lower epitope contacts to a given residue can be weighted differently. The integration of HSE showed to improve previously obtained results.

Lastly, I present an attempt to predict the HIV-1 Protease specificity. As the protease is essential for the life cycle of the HIV virus, the protease is of great interest as an target for the rational design of drugs against HIV. We show that it is possible to predict the specificity of the HIV protease with a high performance. In the process we also identified new possible cleavage sites which will further be verified experimentally in the lab.

In summary, the thesis presented in this work has greatly contributed to the development of new tools in bioinformatics that will hopefully aid in future scientific discoveries.

Dansk resumé

I de seneste årtier har vi set en eksponentiel vækst af biologisk sekvens data. Omkostningerne for DNA sekventering er faldet betydeligt siden annonceringen af det første sekventerede genom og nye sekventerede genomer offentliggøres næsten hver uge. Offentligt tilgængelige genetiske sekvens databaser, som for eksempel GenBank, er vokset betydeligt i størrelse og GenBank indeholder nu mere end 132 millioner sekvenser. Tilsvarende indeholder PDB (Protein Data Bank) på nuværende tidspunkt mere end 71.000 eksperimentelt bestemte strukturer af nukleinsyrer, proteiner og nukleinsyre / protein komplekser. Hvis man sammenligner mængden af eksperimentelt verificerede proteiner med mængden af DNA-sekvenser, så ses det tydeligt at der er en enorm overrepræsentation af DNA-sekvenser. Det akademiske og industrielle forskning samfund må derfor stole på struktur forudsigelser i stedet for at vente på de tidskrævende eksperimentelt bestemte struktur data.

Denne afhandling beskriver udviklingen af to nye værktøjer til at studere sådanne genetisk sekvens data. NetSurfP blev udviklet til at forudsige overflade tilgængeligheden af aminosyrer i aminosyre sekvenser. Kendskab til graden af overflade eksponering af en aminosyre er værdifuld og har været anvendt til at øge forståelse for en række forskellige biologiske problemer, blandt andet protein-protein interaktioner, forudsigelse af epitoper og aktive sites. Efter NetSurfP blev NetTurnp udviklet til forudsigelse af β-turns, og med hjælp af sekundær struktur og overflade tilgængeligheds forudsigelser fra NetSurfP, gav det en bedre forståelse og forbedring af tidligere opnåede resultater for forudsigelse af β-turns.

β-turns er meget interessante idet de er den mest udbredte form for turn strukturer, og omkring 25% af alle aminosyrer i protein strukturer er beliggende i et β-turn.

Indenfor bioinformatikken er hastighed og præcision en vigtig faktor, og derfor forventes de udviklede værktøjer at returnere et resultat på en hurtig og effektiv måde. Vores måde at løse dette problem på var at forudberegne protein sekvens data. I øjeblikket er der mere end 500.000 protein sekvenser i den lokale cache.

I forhold til overflade eksponering omhandlede det tredie projekt forudsigelse af diskontinuerlige B-celle epitoper. Her blev Half Sphere Exposure (HSE) integreret i et allerede eksisterende værktøj. HSE er en mål til forklaring af overflade eksponering hvorved de øvre og nedre epitop kontakter kan vægtes forskelligt. Integrationen af HSE viste sig at forbedre tidligere opnåede resultater.

Endeligt præsentes et forsøg på at forudsige HIV-1 proteasens specificitet. Proteasen er af altafgørende betydning for HIV-virussens livscyklus, og er derfor af stor interesse som et mål for rationelt design af lægemidler imod HIV. Vi viser, at det er muligt at forudsige HIV proteasens specificitet med høj præcision. I processen har vi yderligere identificeret nye mulige kløvningssteder, der efterfølgende vil blive kontrolleret eksperimentelt i laboratoriet.

Sammenfattende har PhD arbejdet præsenteret i denne afhandling i høj grad bidraget til udviklingen af nye værktøjer indenfor bioinformatikken, som forhåbentlig vil støtte fremtidige videnskabelige opdagelser.

Acknowledgements

It has been a great pleasure to work as a PhD student at the Center for Biological Sequence (CBS). Professor Søren Brunak deserves great thanks for providing so friendly and scientific-stimulating atmosphere. People at CBS are always ready to help when you need it or to just have a cup of coffee in the couch and a nice talk. To all the CBSians, I thank you for being a part of my life throughout the last three years.

Special thanks to my supervisors Claus Lundegaard and Thomas Nordahl Petersen. I will forever be grateful that you took me under your wings and believed in me. Thank you for the many hours of valuable discussions and for all the things you have taught me within the field of bioinformatics and about being a researcher. Thank you for pushing me when I needed to be pushed, and for motivating me when I needed to be motivated. This has definitely been a great journey for me.

To my family, especially my mom and dad, I can never thank you enough for being who you are ! And to my grandfather who has his 96 years birthday today. You are such an inspiration towards how to approach life, and I can only hope that I have inherited some of your spirit.

During my PhD I have had the pleasure of being a member of two research groups, the vaccine group and the functional human variation which has joint meetings with the metagenomics group. In my main group, the vaccine group Ole has managed to find so many nice people, who I really have enjoyed my company with: Claus, Edita, Hao, Ilka, Leon, Massimo, Mette, Morten, Nico, Ole, Pernille, Shiela and Stranzie. In the FHV group: Ramneek, Agata, Juliet, Kasper, Morten, Thomas x 2. In the metagenomics group, Thomas, who still falls for that old trick: ”Look, there is a bird!” (and then he loses in tablefootball ;)), Henrik-Bjørn, Ida, Josef, Henrik for also being my co-supervisor for the first months, Marcello, Simon, Thomas, Olga and Ulrik.

My gratitude to Martin Willemoës, for offering me a desk at Copenhagen University with a very short notice.

My office-mates at CBS, Leon, Massimo and Ulrik for coping with my weird taste of music, showing me all the small tricks to Unix, LaTeX, Python etc. and most of all for all the nice time we have had.

I am grateful to the people who proofread this thesis: Bo, Claus, my girlfriend Madara and Thomas. Your comments and corrections have been really valuable and appreciated.

To all my colleagues at CBS, especially Agata, Aron also for his excel-

lent LaTeX template, Daniel, Edita, Eva, Greg, Juliet, Ida, Kasper, Kirstine, Marcelo, Massimo, Nico, Nicolai, Nils, Oksana, Simon, Sonny, Tejal, Stranzie, and all the others who make this place so nice.

To my daily lunch buddies Bille, Jesper and Thomas.

To my close friends: Bo, Casper, Chico, Hannah, Jesper and Naiwai. You all know why you are here !

To the sysadmin for all the times you have been such a help for me, Hans-Henrik, John, Kristoffer and Peter.

To the administration for being who you are, and not just admin people. Annette, Dorthe, Lone, Louise, Malene and Stine.

To Anna and Mustafa.

To Christian Kromann. My biology teacher from high-school. You were such a huge inspiration and motivational factor for me to pursue my future in biology.

If I had to list everyone that means something to me, this would be longer than my thesis. If you are not on the list, it does not mean that you are forgotten.

And of course my girlfriend Madara, who has been even more patient than I could have imagined. Thank you for your cheerful mood, for motivating me, for your help reading the thesis, understanding me and for your patience. I guess I have not been fun the last months :)

Papers included in the thesis

- **Paper I: Bent Petersen**, Thomas Nordahl Petersen, Pernille Andersen, Morten Nielsen and Claus Lundegaard.
  A generic method for assignment of reliability scores applied to solvent accessibility predictions. *BMC Structural Biology*, 9:51, 2009.

- **Paper II: Bent Petersen**, Claus Lundegaard and Thomas Nordahl Petersen.
  NetTurnP – Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features. *PLoS ONE*, 5(11):e15079, 2010.

Abbreviations

AIDS	Acquired Immuno Deficiency Syndrome
ANN	Artificial Neural Networks
AROC	Area under the Receiver Operating characteristics Curve
ASA	Accessible Surface Area
FN	False Negative
FP	False Positive
HIV	Human Immunodeficiency Virus
HSE	Half Sphere Exposure
HMM	Hidden Markov Models
LAH	Los Alamos HIV sequence database
MCC	Matthews Correlation Coefficient
NN	Neural Network
PCC	Pearsons Correlation Coefficient
PPV	Predicted Positive Value
PR	HIV-1 Protease
PS	Propensity Scores
PSSM	Position Specific Scoring Matrix
ROC	Receiver Operating characteristics Curve
SNP	Single Nucleotide Polymorphism
SVM	Support Vector Machine
RSA	Relative Surface Area
TN	True Negative
TNT	β-turn / not-β-turn
TP	True Positive

List of Figures

List of Tables

General Introduction

Chapter 1

Introduction

1.1 Motivation and local structural features

The field of bioinformatics has exploded within the last 15 - 20 years, and the amount of data available to the public in huge databases like for example GenBank (Benson et al. (2005)) is increasing considerably. GenBank is a genetic sequence database, which contains all public available annotated DNA sequences. GenBank currently (February 15, 2011) contains more than 132 million sequences with more than 124 billion nucleobases, see Figure 1.1. It is a huge amount of data, and according to GenBank the size of the database doubles approximately every 18 months.

The Protein Data Bank (PDB) (Berman et al. (2000)) is an archive containing information about experimentally determined structures of nucleic acids, proteins and nucleic acid/protein complexes. The vast majority of structures in the database are proteins which have been solved using X-ray crystallography, whereas other techniques, such as NMR and electron microscopy, are also contributing to the growth. There are currently (March 2011) more than 62,200 X-ray solved structures, with a total of 71,635 three-dimensional structures in PDB. The database is growing with around 7,000 - 8,000 structures per year, see Figure 1.2.

Comparing the amount of public available DNA sequences with the amount of experimentally verified proteins, there is a huge over-representation of DNA sequences and the amount of genomic data is clearly growing faster than the rate of experimentally determined three-dimensional structures. In order for the academic and industrial research community to gain use of all the sequence data, they have to rely on structure predictions instead of waiting for the time consuming experimentally determined structural data.

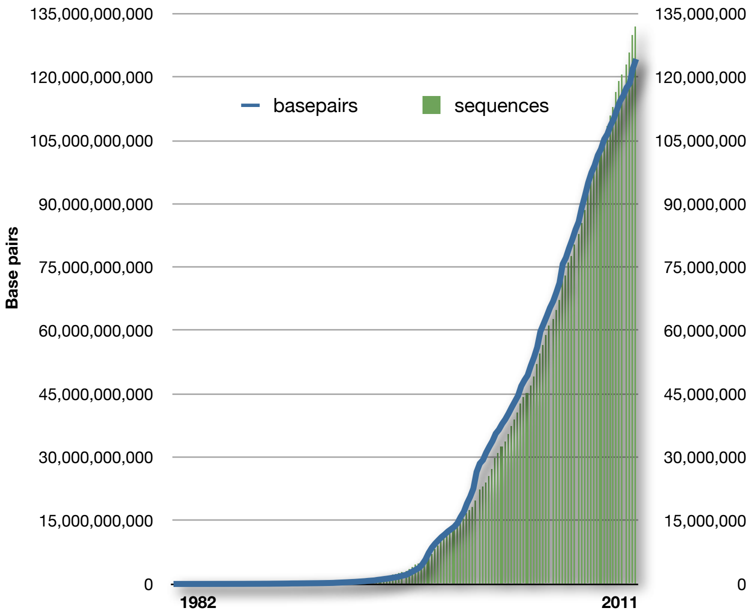


Figure 1.1. The figure shows the growth of GenBank in the period 1982-2011. The green area is the amount of sequences, and the blue line is the amount of base pairs.

Machine-learning algorithms have proven to be very useful as prediction tools, and have been used within many different scientific areas. CBS has for a long time contributed with many biological prediction tools of which several are publicly available. A few of them are: Signal peptide prediction - SignalP (Dyrl Bendtsen et al. (2004)), N-Glycosylation sites in human proteins - NetNGlyc (Gupta et al. (2004)) and MHC peptide binding - NetMHC (Lundegaard et al. (2008)). Different machine-learning algorithms exist where the most widely known are: Support Vector Machines (SVM), Hidden Markov Models (HMM) and Artificial Neural Networks (ANN). In the work presented in this thesis, ANN has been the main vehicle driving the projects, and therefore only ANN is described, see Section 1.3.

The driving force behind the projects described in this thesis was to develop tools for prediction of surface exposure and β-turn occurrence, which are useful for the broad academic and scientific research community, but keeping in mind the actual use and usability of a prediction tool by other people is difficult to foreseen.

The content of the thesis reflects the previously mentioned driving force in the six following chapters: Chapter 1 is the general introduction. Chapter 2 describes the first developed tool NetSurfP, which predicts the surface accessibility and secondary structure of amino acids in an amino acid sequence. The

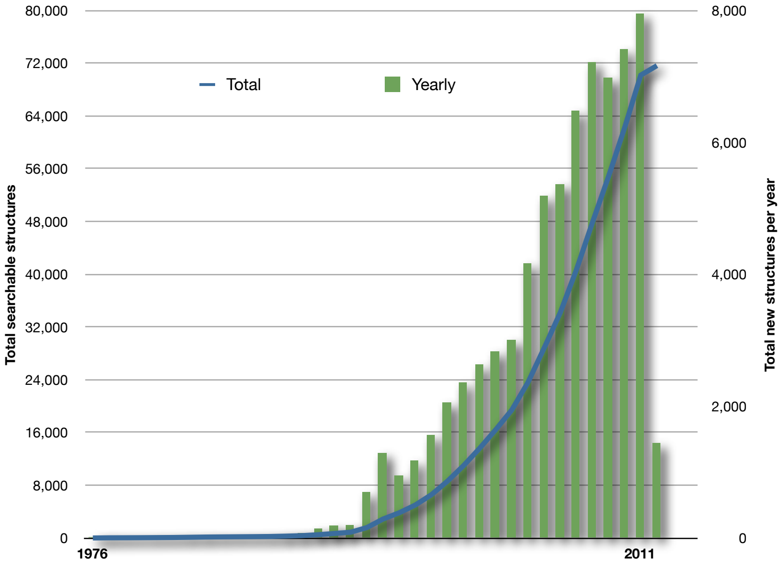


Figure 1.2. The figure shows the growth of Protein Data Bank (PDB) in the period 1976 -2011. The green bars show the amount of new sequences per year, and the blue line shows the total number of PDB structures.

secondary structure predictor was not developed during this work, instead an already existing in-house tool was integrated in the NetSurfP tool. Paper I describes the development of that tool. The manuscript has been reformatted for the thesis, but is otherwise identical to the published version. Chapter 3 describes the second developed tool NetTurnP, which predicts whether or not an amino acid is located in a β-turn. NetTurnP is also able to predict the nine β-turn subtypes. Paper II describes the development of NetTurnP. Again the manuscript has been reformatted for the thesis, but is otherwise identical to the published version. Chapter 4 describes a caching project which was initiated to speed-up the two previously mentioned tools. Chapter 5 describes the project performed at Copenhagen University with a goal to predict the substrate recognition sites for the HIV-1 protease. Chapter 6 describes the attempt to improve an already existing tool named DiscoTope, which predicts discontinuous B-cell epitopes.

When a tool is developed, it is difficult to speculate in which context it will be used. NetSurfP is a good example of that. Since the publication and launch of NetSurfP as a web-server, it has been used extensively within different scientific areas. Supplementary Table 2 lists papers (published until and including March 2011) citing or mentioning the use of NetSurfP. As it can be seen NetSurfP has been used for various purposes, for example veterinary immunology and immunopathology, proteome research, antigenic epitopes prediction and structural biology. NetSurfP predictions has also been implemented in the following in-house tools: CPH-models (Nielsen et al. (2010)), Epipe (Blicher et al. (2010)), NetdiseaseSNP (personal communication, not yet published) and NetTurnP (Petersen et al. (2010)).

1.2 Position Specific Scoring Matrices

A Position Specific Scoring Matrix (PSSM), is a substitution matrix of length 20 times the length of a protein sequence. Each element is a log-odds score that indicate weather an amino acid substitution at a certain position is more or less likely compared to what is expected. The substitution matrix is thus able to capture evolutionary information from a family of near and remote sequence homologs. This is in contrast to the generic family BLOSUM (Henikoff and Henikoff (1992)), substitution matrices (BLOSUM30, 50, 62 and others) that are symmetric with a dimension of 20 times 20 and therefore lack sequence specific information.

It is generally known that the overall three-dimensional structure is more conserved than the primary sequence (Illergård et al. (2009)). This means that structural features like secondary structure, surface exposure and active site tend to be preserved even in among remote sequence homologs. A result of this is that when a large family of sequence homologs have been found via PSI-BLAST (Altschul et al. (1997)), we would expect that they share many structural similarities and thus this sequence information is captured by the PSSM.

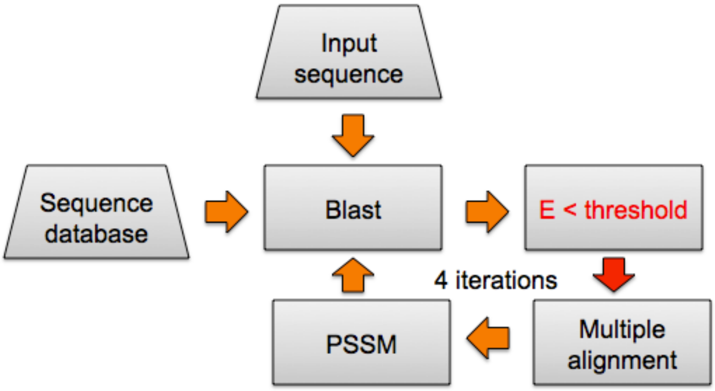
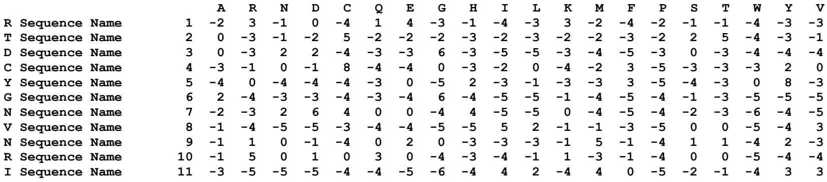


Figure 1.3. Illustration of the process for creating a PSSM.

In the work performed in this PhD thesis, PSSMs were created using the iterative PSI-BLAST program. Query sequences were blasted for four iterations against a local copy of the National Center for Biotechnology Information (NCBI) non-redundant (nr) sequence database, which for speed purposes had been homology-reduced using CDHIT (Huang et al. (2010)) to less than 70% identity. An E-value cut-off of 1×10^{-5} was used. In the process of creating a PSSM, first a normal pairwise local alignment search is is done, using BLOSUM62 as substitution matrix. Secondly, a PSSM is calculated from the multiple alignment and used as the new substitution matrix for the next round. The process is shown in Figure 1.3.



```
                              A   R   N   D   C   Q   E   G   H   I   L   K   M   F   P   S   T   W   Y   V
R Sequence Name           1  -2   3  -1   0  -4   1   4  -3  -1  -4  -3   3  -2  -4  -2  -1  -1  -4  -3  -3
T Sequence Name           2   0  -3  -1  -2   5  -2  -2  -2  -3  -2  -3  -2  -2  -3  -2   2   5  -4  -3  -1
D Sequence Name           3   0  -3   2   2  -4  -3  -3   6  -3  -5  -5  -3  -4  -5  -3   0  -3  -4  -4  -4
C Sequence Name           4  -3  -1   0  -1   8  -4  -4   0  -3  -2   0  -4  -2   3  -5  -3  -3  -3   2   0
Y Sequence Name           5  -4   0  -4  -4  -4  -3   0  -5   2  -3  -1  -3  -3   3  -5  -4  -3   0   8  -3
G Sequence Name           6   2  -4  -3  -3  -4  -3  -4   6  -4  -5  -5  -1  -4  -5  -4  -1  -3  -5  -5  -5
N Sequence Name           7  -2  -3   2   6   4   0   0  -4   4  -5  -5   0  -4  -5  -4  -2  -3  -6  -4  -5
V Sequence Name           8  -1  -4  -5  -5  -3  -4  -4  -5  -5   5   2  -1  -1  -3  -5   0   0  -5  -4   3
N Sequence Name           9  -1   1   0  -1  -4   0   2   0  -3  -3  -3  -1   5  -1  -4   1   1  -4   2  -3
R Sequence Name          10  -1   5   0   1   0   3   0  -4  -3  -4  -1   1  -3  -1  -4   0   0  -5  -4  -4
I Sequence Name          11  -3  -5  -5  -5  -4  -4  -5  -6  -4   4   2  -4   4   0  -5  -2  -1  -4   3   3
```

Figure 1.4. A Position Specific Scoring Matrix for an 11 amino acid long protein sequence. Column 1: amino acid in the sequence, column 2: sequence name, column 3: amino acid number, column 4-23: log-odds scores for all twenty amino acids.

Figure 1.4 illustrates an example of a PSSM for a sequence of 11 amino acids. It can be noticed that the PSSM consists of vectors of the size 20 and each vector is specific for a position in the amino acid sequence. The scores in the matrix are log-odds scores between the observed q_{ij} and expected $p_i p_j$ pair frequency of amino acids (i, j), thus:

$$S_{ij} = 2log_2(\frac{q_{ij}}{p_i p_j}) \tag{1.1}$$

The scores in the PSSM are shown as positive or negative integers. A positive score indicates that the given amino acid substitution occurs more frequently than expected, and a negative score indicates that the given amino acid substitution occurs fewer times than expected. To give an example, the isoleucine at position 11 in the sequence in Figure 1.4 is matched to a glycine with a score of -6, indicating that this substitution is less likely to be observed. However, the match score for a methionine is 4, which is the same score as for an isoleucine itself. As a comparison the BLOSUM62 substitution score for isoleucine to methionine is 1.

1.3 Artificial neural networks

Artificial Neural Networks (ANN) is a machine learning technique inspired by the architecture and functionality of the human brain. The concept of ANN was originally invented in the 1940 and has gained popularity due to its ability to solve non-linear pattern classification problems. Within biological sequence analysis ANN has successfully been used for various prediction problems, for example, secondary structure, glycosylation, phosphorylation, peptide cleavage, surface accessibility and epitope prediction.

The ANN that has been used during this work is a standard feed-forward multilayer network (Rumelhart et al. (1986)), where neurons are arranged in layers and information flows from one layer to the next. The first layer is the input that will contain the biological sequence data of interest. Each neuron in the input layer is connected to the neurons in a hidden layer, which is further connected to the neurons in the output layer. All connections between the neurons are called synapses and each of them are associated with a weight. The number of output neurons in the network depends on the problem being analysed. For a classification problem, which can be solved with yes/no, there are normally two output neurons. We also have threshold neurons connected to each hidden and output neuron. For the ANN's used in this work, these neurons have a constant value of -1, and the weights are updated as all other weights. One can think of the threshold neurons as a value that is added to for example a hidden neuron and thus determines if that hidden neuron should "fire" or not. For simplicity, the threshold neurons are not shown in Figure 1.5 that illustrates a conventional feed-forward neural network with one hidden layer.

Input data has to be encoded such that it can be interpreted by the ANN. In order to encode the input, different encoding schemes can be used. As an example sparse encoding and the use of Position Specific Scoring Matrices (PSSM) (previously described in Section 1.2) can be mentioned. Only PSSM has been used in this work.

In sparse encoding each amino acid is represented as either absent (0) or present (1) in string of 21-bit binary values. For example, an alanine is encoded as 100000000000000000000 and a cysteine as 010000000000000000000. The 21st bit represents if the N- or C-terminal of a sequence is reached i.e. encoded amino acid is outside the termini of the protein sequence, which can happen when a window of more than one amino acid is feed into the ANN. Typically a window of several amino acids are used, where predictions are made for the amino acid in the center of the window. Sparse encoding is used when the information about the particular amino acid is important. That could, for example, be in the prediction of cleavage sites or glycosylation.

When a PSSM encoding scheme is used, the information about the actual amino acid is lost. Instead the evolutionary information is presented to the

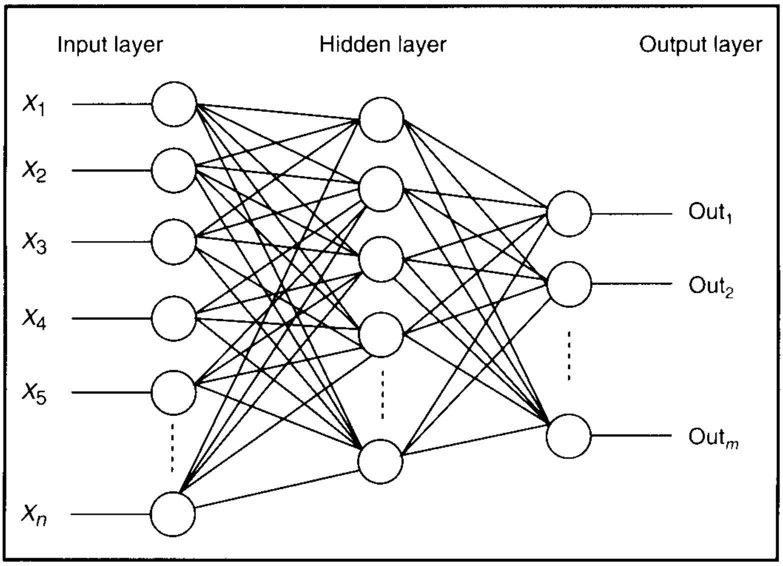


Figure 1.5. The figure shows a conventional feed-forward neural network. The ANN has an input layer with n neurons, a hidden layer with x neurons and m output neurons. Picture courtesy of V. Venugopal[1].

ANN in form of a 21-bit vector consisting of log-odds scores for all amino acids, as described in Section 1.2

Supervised learning was used to train the ANN's in this work. It means that the algorithm was provided with both input and target values. By iterating over several training rounds or epochs, the ANN updates the weights by use of a gradient descent method where the error is back-propagated. The most often used error function for classification problems is the sum of squared differences which is presented in Equation 1.2:

$$E = \frac{1}{2}\sum_i (t_i - o_i)^2 \tag{1.2}$$

Where o_i is the output from the ANN and t_i is the target value for the training example.

Mathematically a hidden neuron can be represented as an input function h_j, an activation function f_x and an output function H_j. Input to a hidden neuron H_j can be calculated as:

$$h_j = \sum_k v_{jk} I_k \tag{1.3}$$

[1] Venugopal, V., Baets, W. (1994). Neural networks and statistical techniques in marketing research: A conceptual comparison. Marketing Intelligence Planning, 12(7), 30-38.

Where I_k is the received input and v_{jk} is the weight on the input k to the hidden neuron j. The output from the hidden neuron H_j is:

$$H_j = f(h_j) \tag{1.4}$$

Where $f(x)$ most often used for non-linear problems is the sigmoidal function:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{1.5}$$

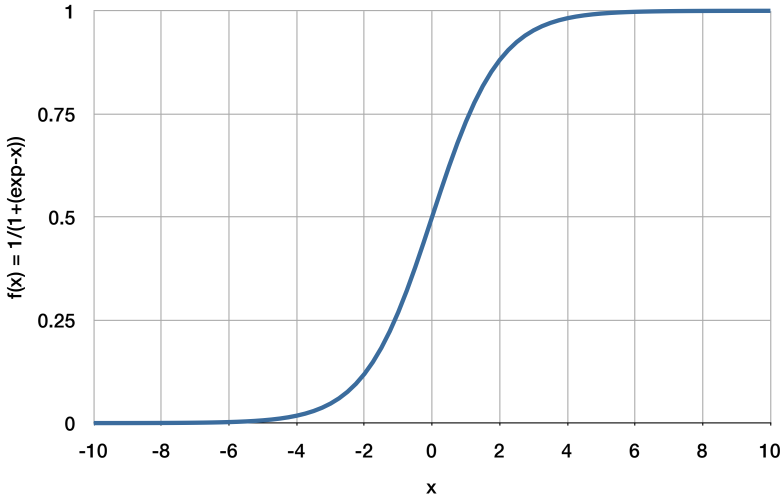


Figure 1.6. The figure shows the sigmoidal function, which is used as activation function for the neural network.

Figure 1.6 shows a plot of a sigmoidal function, which is a graphical representation of Equation 1.6. It can be noticed that the highest sensitivity is in the region, where values on X-axis are close to 0. Beyond or below ± 4, the function change only slightly.

1.3.1 Training stopping criteria

TRaining of neural networks requires a procedure for stopping the training at the right time. An example of a neural network training can be seen in Figure 1.7. The figure shows a simplified training example as a graph with number of epochs (training rounds/steps) on the X-axis, and the error of that particular epoch on the Y-axis. The green curve illustrates the test set and the red curve illustrates the training set. For a real training situation the curves would be more fluctuating, see Figure 1.8. The time point illustrated with the vertical dotted line (Figure 1.7) represents the epoch, where the error on the test set (green curve) starts to increase, while the error on the training set (red curve) continues to decrease. The possible reason for this might be that an over-fitting has occurred.It means that the algorithm might have lost its ability to generalize and instead learns the training set data better and better, which shows as a continuously decreasing error. In other words, the algorithm has become too specialized. As long as we have a training, test and an evaluation data set that are properly homology reduced, we can be reasonable sure that overfitting do not occur. If only a training and test set were used then one should be cautious if testing is done more than once. In these cases it is preferable to have a parameter weight ratio above one. For the ANN's the ratio can be calculated as shown in Equation 1.6 , where 'win' = window's size, 'hid' = amount of hidden units, 'out' = number of output neurons. Here the number of threshold neurons are 'hid'+'out'.

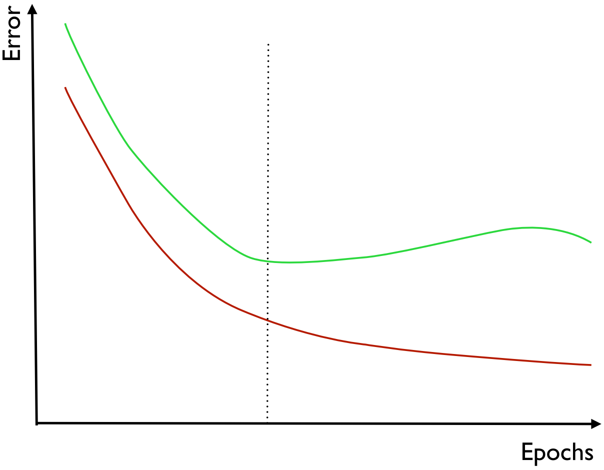


Figure 1.7. The green curve on the figure is the error of the test set, and the red curve is the error of the training set. The dotted line illustrates the time point where the error of the training set increases while the test error continues to decrease.

$$\frac{\text{Examples}}{\text{Parameters}} = \frac{\text{Examples}}{(\text{win} \times \text{input} \times \text{hid}) + (\text{hid} \times \text{out}) + \text{hid} + \text{out}} \tag{1.6}$$

To avoid over-fitting in the neural networks trained in the work described in this thesis, two approaches have been applied. The first approach applies directly to the neural network training. During the training, the settings and weights are saved for the epoch giving the highest test correlation i.e. stopping criteria. Matthews Correlation Coefficient (MCC) is used for the classification networks (for example, exposed or not exposed) and Pearsons Correlation Coefficient (PCC) for the real-value networks (for example, exposure as values between 0 (buried) and 1 (fully exposed)). An example of an epoch where the test MCC is highest and the error is lowest, can be seen in Figure 1.9. This figure shows zoomed-in plots of Figure 1.8, representing epochs in a range between 100 and 150. Epoch number 127 is in this training giving the lowest error for the test-set (green curve, figure to the left) and in the same epoch the test-set also has the highest MCC (green curve, figure to the right). Weights and settings are therefore saved at this epoch, which is further used for the calculation of the performance of the trained method.

The other approach used to avoid over-fitting is by introducing a totally independent dataset, which is used for the final evaluation of the neural networks, named the evaluation dataset. This dataset has a sequence identity which is less than 25% compared to the training and test datasets. The evaluation performance thus gives a good estimate of how well the method is performing on new data.

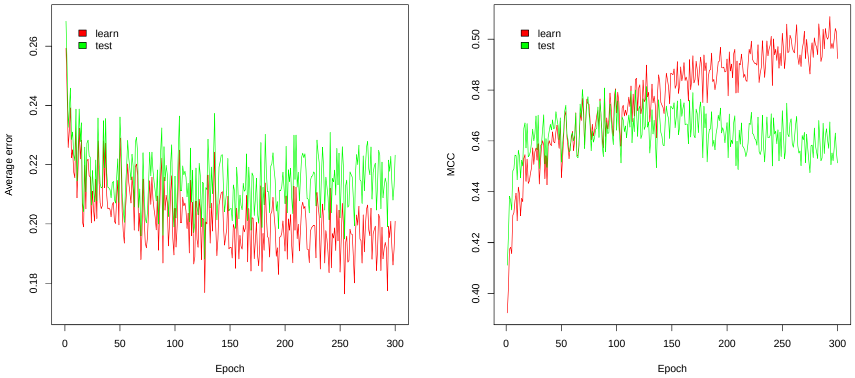


Figure 1.8. The figure to the left represents the error per amino acid for a neural network running 300 epochs, and the figure to the right illustrates the performance, measured in MCC. The green curve is the test set, and the red curve is the training set.

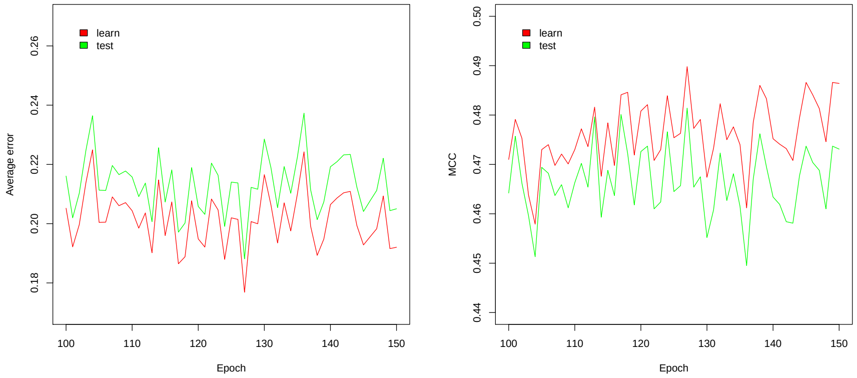


Figure 1.9. The two figures are zooms of figure 1.8 from epoch 100 to 150. Figure to the left shows that the lowest error per amino acid on the test set is at epoch 127, and figure to the right represents the corresponding MCC, which also is highest at epoch 127. The green curve is the test set, and the red curve is the training set.

1.3.2 Cross-validation

Cross-validation is a training procedure used to assess how well the results of an analysis generalize to an independent dataset. In the work presented here, a 10-fold cross-validation has been used, unless otherwise stated. This implies that the dataset is randomly partitioned into ten subsets. One subset is retained from the training (denoted the test set) and the remaining nine subsets are used for the learning/training (denoted the training set). This procedure is then repeated as many times as there are folds, each time retaining a new subset from the training. When all folds have been used as test set, the results are averaged to give a single measure of the test performance. The advantage of this method is, that all data is used for both testing and training, and that each data partition is used for testing only once.

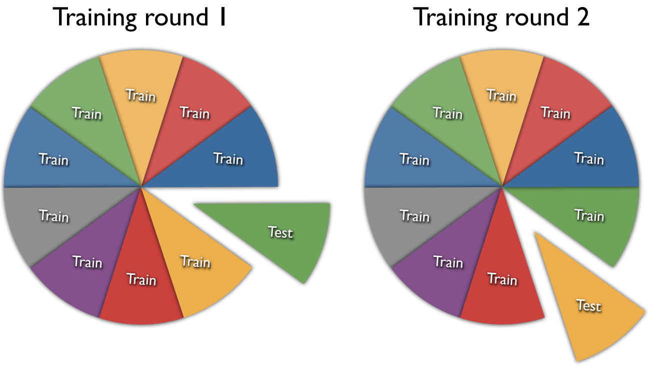


Figure 1.10. The figure illustrates two rounds of a 10-fold cross-validation. In the first round the green subset is used for testing, and the nine remaining subsets for training, whereas in the second round the green subset is now included in the training, and the yellow is now used for testing. This procedure is repeated until all ten subsets have been used for testing.

When all weights and settings have been optimized based on the 10-fold average test performance, the method is benchmarked using an evaluation dataset. As previously mentioned, we used 25% sequence identity cut-off and the evaluation is only done once on the trained model.

1.4 Sequence logos

SEquence logos are graphical representations of the patterns in a multiple sequence alignment, developed by Tom Schneider and Mike Stephens (Schneider and Stephens (1990)). From a sequence logo one can determine the information content and the relative frequency of the letters at every position in the sequence alignment. They are often used to visualize motifs in DNA/RNA or protein sequences. In the work described in this thesis sequence logos are only used to examine protein sequences and the protein sequence assignments.

At each position in the sequence, the information content is indicated as the total height of a stack of letters and the height of each letter represents the frequency of that letter. The total height of the stack is represented in bits and is calculated using Claude Shannon's measure of uncertainty (Shannon (1948)). The higher the stack of the letters, the more conserved this position is across all the sequences. The entropy or Shannon entropy (H) at a given position i is defined as shown in Equation 1.7:

$$H(i) = -\sum_{a} p_a \log_2(p_a) \tag{1.7}$$

Where a is the set of all twenty amino acids and p_a is the probability of seeing the given amino acid a at that position. The information content, I_c, for position i is defined as in Equation 1.8. The value of the information content ranges from 0 (no conservation; all amino acids are equally probably at that position) to $\simeq 4.3$ (full conservation; one single amino acid is observed at that position) in the case of amino acids. For nucleotides, the maximum information content is: $I_c(i)= log_2(4) = 2$.

$$I_c(i) = log_2(20) - H(i) \tag{1.8}$$

In logo plots amino acids are normally coloured according to their physiochemical properties, in this work with the following colours:

- Acidic [DE]: **Red**
- Basic [HKR]: **Blue**
- Hydrophobic [ACFILMPVW]: **Black**
- Neutral [GNQSTY]: **Green**

Figure 1.11 shows a logo plot for a part of the protein sequence for an acetylesterase from Aspergillus aculeatus. Two amino acids, which are important for the function of the acetylesterase, are clearly captured by the sequence logo and can be seen as amino acid D209 and H212. Various tools are available on-line to create sequence logos. Weblogo is an example of a web based application for generation of sequence logos: http://weblogo.berkeley.edu/ (Crooks et al. (2004)). Slogo is another example of a web based tool for generation of sequence logos: http://www.cbs.dtu.dk/~gorodkin/appl/slogo.html (Gorodkin et al. (1997))

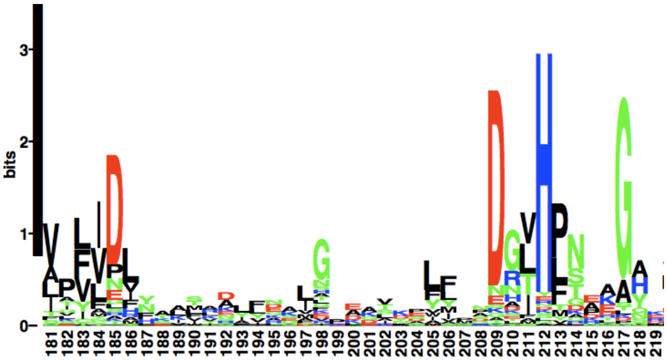


Figure 1.11. The figure shows a logo plot for a part of the sequence for Rhamnogalacturonan acetylesterase, short name RGAE, which is an acetylesterase from Aspergillus aculeatus. Two of the important amino acids, D209 and H212, are clearly captured by the sequence logo and can be seen as amino acid D209 and H212, which are a part of a catalytic triad (last amino acid in active site is Ser9).

1.5 Performance Measures

1.5.1 Matthews Correlation Coefficient

THe Matthews Correlation Coefficient (Matthews (1975)) (MCC) measures the quality of a binary prediction. A binary prediction could be to predict whether or not an amino acid is exposed (will be discussed in Section 2.1), or whether or not an amino acid is located in a β-turn (will be discussed in Section 3.1) . In this type of predictions the following four outcomes are possible, here using prediction of β-turn as an example:

- True Positive (TP): The amino acid has been correctly predicted to be located in a β-turn.
- False Positive (FP): The amino acid has been falsely predicted to be located in a β-turn.
- True Negative (TN): The amino acid has been correctly predicted not to be located in a β-turn.
- False Negative (FN): The amino acid has been falsely predicted not to be located in a β-turn.

Matthews correlation coefficient can be in the range of -1 to 1, where 1 is a perfect correlation and -1 is a perfect anti-correlation. A value of 0 indicates no correlation.

$$\mathrm{MCC} = \frac{\mathrm{TP\,x\,TN} - \mathrm{FP\,x\,FN}}{\sqrt{((\mathrm{TP}\ +\ \mathrm{FN})\,\mathrm{x}\,(\mathrm{TN}+\mathrm{FP})\,\mathrm{x}\,(\mathrm{TP}+\mathrm{FP})\,\mathrm{x}\,(\mathrm{TN}+\mathrm{FN}))}} \tag{1.9}$$

Q_{total} is the percentage of correctly classified residues, also called the prediction accuracy

$$\mathrm{Q_{total}} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}} \tag{1.10}$$

PPV is the Predicted Positive Value, also called the precision or Q_{pred}.

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}\,\mathrm{x}\,100 \tag{1.11}$$

Sensitivity is also called recall or Q_{obs}, and is the fraction of the total positive examples that are correctly predicted.

$$\mathrm{Sensitivity} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}\,\mathrm{x}\,100 \tag{1.12}$$

Specificity is the fraction of total negative examples that are correctly predicted.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \,\text{x}\, 100 \tag{1.13}$$

The above mentioned performance measures are all threshold dependant and in this work a threshold of 0.5 has been used. AUC is a threshold independent measure, and it is calculated from the ROC curve which is a plot of the sensitivity against the False Positive rate = FP/(FP + TN) (1-specificity). An AUC value above 0.7 is an indication of a useful prediction and a good prediction method achieves a value $>$ 0.85 (Lund et al. (2005)).

1.5.2 Pearson's Correlation Coefficient

PEarson's correlation coefficient (Press et al. (1992)) (PCC), also called Pearson's r is a linear correlation coefficient, and is a measure of the linear correlation between two variables; here a, actual and p, predicted. As for the MCC values fall between -1 and 1, where a value of 1 is a perfect correlation and -1 is a perfect anti-correlation.

$$PCC = \frac{\sum_i (a_i - \bar{a})(p_i - \bar{p})}{\sqrt{\sum_i (a_i - \bar{a})^2) \sum_i (p_i - \bar{p})^2}} \tag{1.14}$$

Chapter 2

NetSurfP - in-house tool and webserver

2.1 Surface accessibility

Surface accessibility is a measure for describing to what degree an amino acid is accessible to a solvent surrounding the protein, where the solvent usually is water. The measure is also called Accessible Surface Area (ASA) and is given in Å^2. The area is calculated by rolling a sphere with the size of a water molecule over the protein surface (Connolly (1983)) as illustrated in Figure 2.1. For comparative and predictive reasons the ASA is often transformed to a Relative Surface Area (RSA), which is calculated as the ASA of a given amino acid residue in the polypeptide chain, relative to the maximal possible exposure of that residue in the center of a tri-peptide flanked with either glycine (Chothia (1976)) or alanine (Ahmad et al. (2003)), see Equation 2.1. The two different maximum ASA values for all amino acids are listed in Supplementary Table S1.

$$RSA = \frac{ASA}{ASA_{MAX}} \cdot 100\% \tag{2.1}$$

Previous studies have shown that prediction of the RSA is significantly more accurate for buried amino acids than for exposed (Dor and Zhou (2007a)). Hydrophobic amino acids are most often buried inside the protein and shielded from the water molecule, and hydrophilic amino acids are at or close to the surface of the protein, thus being exposed to the solvent.

Knowledge of the degree of surface exposure of an amino acid is valuable and it has been used to enhance the understanding of a variety of biological problems, including protein-protein interactions (Jones and Thornton (1997a), Jones and Thornton (1997b)), structural epitopes (Haste Andersen

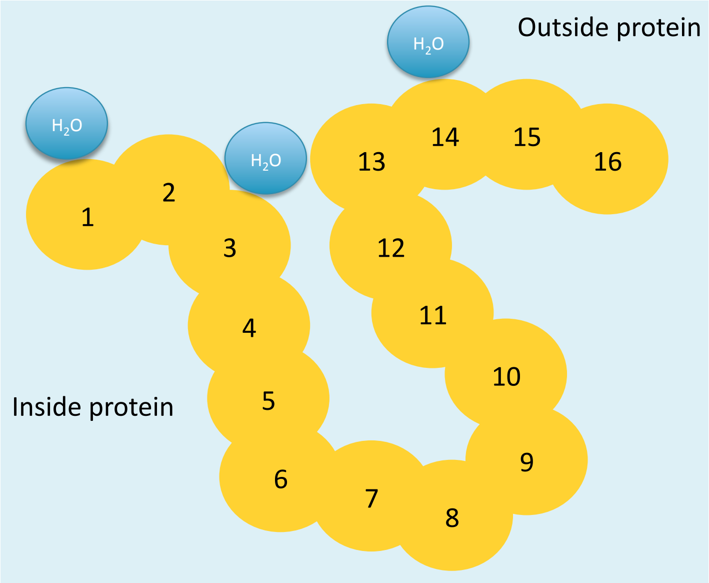


Figure 2.1. The figure illustrates the Accessible Surface Area (ASA) for a protein consisting of 16 amino acids.

et al. (2006)), active sites (Panchenko (2004)), and prediction of disease-related Single Nucleotide Polymorphisms (SNP) (Mooney (2005)).

Several methods for predicting surface accessibility from the primary protein sequence have been developed often inspired by the related field of protein secondary structure prediction as exemplified with (Pollastri et al. (2002)) implemented in (Cheng et al. (2005)). Generally, the best methods involve the use of advanced machine learning algorithms, such as Artificial Neural Networks (ANN) or Support Vector Machines (SVM) combined with evolutionary information. Traditionally surface accessibility has been predicted as a two-state classification categorizing the amino acids as either being buried or exposed using more or less arbitrary cut-offs. Usually an ASA value of more than 25% is defined as exposed. Recently, real value RSA predictors have been developed, thus removing the need to define specific cut-offs (Ahmad et al. (2003)).

The most biologically interesting residues are often exposed due to the reason that they are able to interact with the environment. Unfortunately, highly exposed residues tend to be more difficult to predict than buried amino acids (Ahmad et al. (2003), Dor and Zhou (2007a), Dor and Zhou (2007b)). In order to investigate to what extent the predictability depends on the degree of amino acid exposure, a dataset containing 513 protein sequences (CB513, further explained in Chapter 2.4) was examined using NetSurfP (further explained in Chapter 2.4). Figure 2.2 shows a heat-map of the sensitivity for each amino acid within a given relative surface accessibility range. The figure

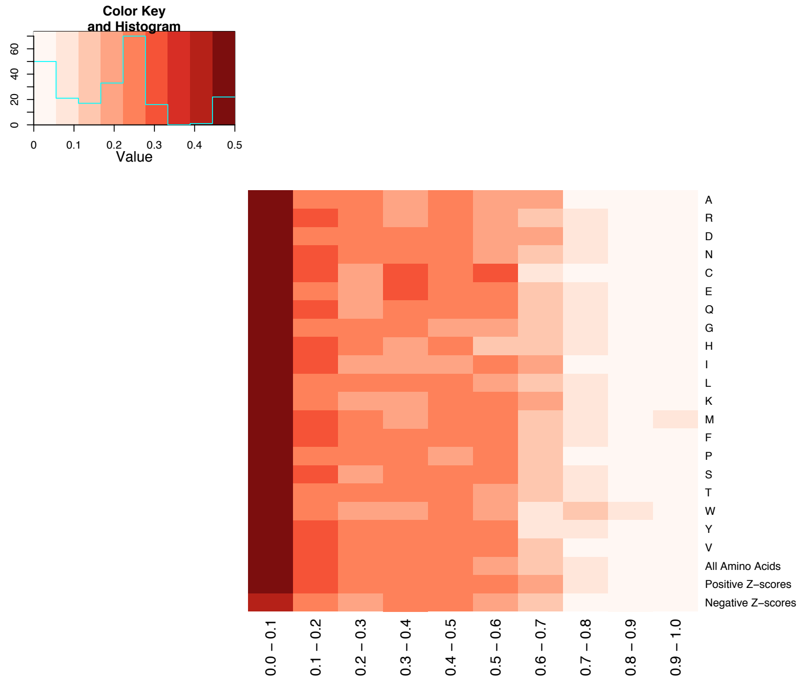


Figure 2.2. The lower right figure shows the sensitivity for each amino acid within given relative surface accessibility ranges. Included are also rows showing the sensitivity for all amino acids, and amino acids with either positive or negative Z-score. In total 230 sensitivity values are shown as coloured squares in the figure. The upper left figure shows the color coding for the sensitivity and also the number of counts for each of the 230 squares shown as a histogram

clearly confirms and demonstrates that predictions with the highest sensitivity fall in the range of 0.0 - 0.6 RSA, and most in 0.0 - 0.2 RSA, meaning that buried amino acids are easier to predict than highly exposed. Moreover, the plot shows that this behaviour is similar for all amino acids.

NetSurfP was developed to predict the surface accessibility of amino acids in an amino acid sequence, and both the buried/exposed classification and RSA value (between 0 and 1) is reported. Simultaneously, the method also predicts the reliability for each prediction in the form of a Z-score. It was found, that data points with a high Z-score had a lower predicted error compared to data-points with a low Z-score. In tests to investigate the validity of the calculated Z-score it was further found that the score could indeed successfully be used to filter out more reliable predictions. It resulted in a significantly better correlation between the predicted and measured values. NetSurfP Z-scores thus enables the identification of the most reliable/unreliable predictions for both buried and exposed amino acids. NetSurfP has been trained on surface exposure measures from DSSP, which also gives measures for complexes if possible. If, for example, a dimer is split into its two monomers, and ASA is calculated separately for each monomer, a bad reflection of the functional parts of the protein is obtained. In a complex some structures could have hydrophobic amino acids at the surface of the monomer, but these probably lie as part of an interface between two protein chains. Therefore instead of using single chains the RSA measures from complexes are used in the work described in this thesis.

2.2 Secondary structure

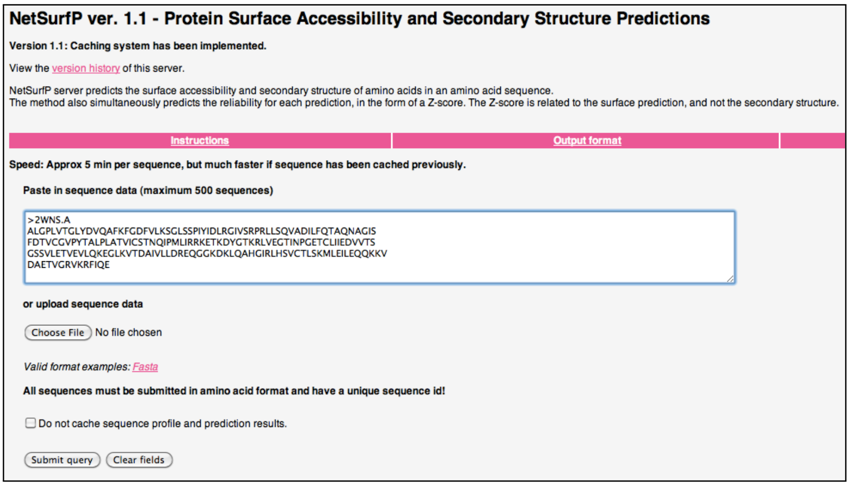
THe secondary structure of proteins is the general three-dimensional form of segments of a polypeptide chain. The work described in this thesis has not been related to secondary structure, but it will nevertheless be shortly introduced because of the implementation of secondary structure predictions in NetSurfP.

Usually three common secondary structures are considered, namely α-helices, β-sheets and turns (or coils). Structures which cannot be classified as one of these three standard classes are usually grouped as 'random coil' or 'other'. DSSP (Kabsch and Sander (1983)) is an algorithm for assigning secondary structure to amino acids in a protein structure. Given a structure file with atomic coordinates of a protein, DSSP identifies the hydrogen bonds of the protein structure and assigns the secondary structure elements depending on the pattern of the hydrogen bonds. DSSP recognizes eight types of secondary structure; G = 3_{10}-helix, H = α-helix, I = π-helix, B = β-sheet, E = Extended strand, T = turn, S = bend and . = loop. These types are usually grouped into larger groups: helix (G, H and I), strand (E and B) and loop (all others).

As mentioned secondary structure prediction has not been performed during this work, it was merely an implementation of an already (unpublished) existing in-house predictor, which NetSurfP used to improve its performance. The secondary structure elements for this predictor were grouped into three classes: The H class comprised by DSSP class H, E class comprised by DSSP class E, and the C class comprised by DSSP classes .,G,I,B,S and T. The performance of the predictor on the CB513 dataset (see Chapter 2.4 for further description) was 81,3% correct predictions, see Table 2.1 for all performance measures.

2.3 NetSurfP in-house tool and web-server

NEtSurfP is both available as an in-house tool and as a web-server. NetSurfP can be accessed from http://www.cbs.dtu.dk/services/NetSurfP/, where the user meets a front-page, which is presented in Figure 2.3. In this example a protein sequence with the name 2WNS.A is submitted and the user has agreed on using the cache (see Chapter 4). After a successful prediction the user receives an output as shown in Figure 2.4.



NetSurfP ver. 1.1 - Protein Surface Accessibility and Secondary Structure Predictions

Version 1.1: Caching system has been implemented.

View the version history of this server.

NetSurfP server predicts the surface accessibility and secondary structure of amino acids in an amino acid sequence.
The method also simultaneously predicts the reliability for each prediction, in the form of a Z-score. The Z-score is related to the surface prediction, and not the secondary structure.

Speed: Approx 5 min per sequence, but much faster if sequence has been cached previously.

Paste in sequence data (maximum 500 sequences)

```
>2WNS.A
ALGPLVTGLYDVQAFKFGDFVLKSGLSSPIYIDLRGIVSRPRLLSQVADILFQTAQNAGIS
FDTVCGVPYTALPLATVICSTNQIPMLIRRKETKDYGTKRLVEGTINPGETCLIIEDVVTS
GSSVLETVEVLQKEGLKVTDAIVLLDREQGGKDKLQAHGIRLHSVCTLSKMLEILEQQKKV
DAETVGRVKRFIQE
```

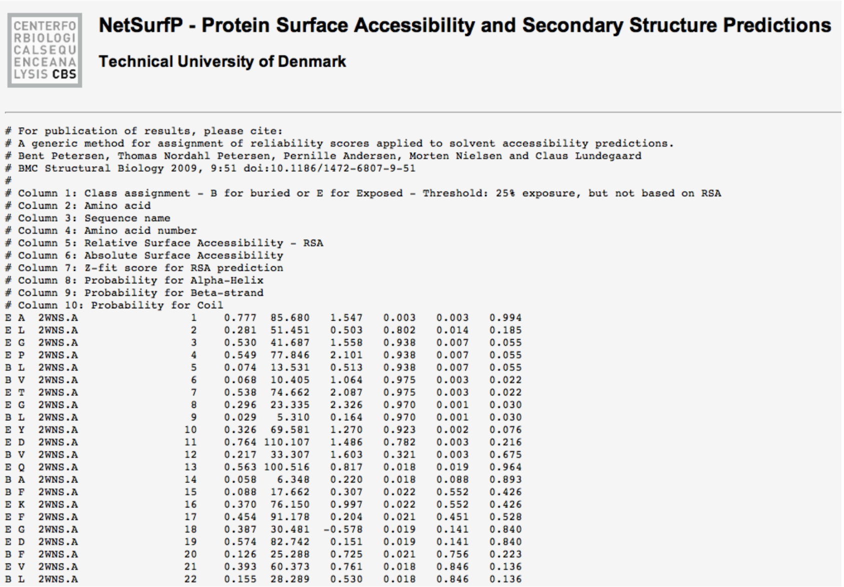
or upload sequence data

Valid format examples: *Fasta*

All sequences must be submitted in amino acid format and have a unique sequence id!

□ Do not cache sequence profile and prediction results.

Figure 2.3. The figure shows the NetSurfP web-server protein sequence submission form.



NetSurfP - Protein Surface Accessibility and Secondary Structure Predictions

Technical University of Denmark

```
# For publication of results, please cite:
# A generic method for assignment of reliability scores applied to solvent accessibility predictions.
# Bent Petersen, Thomas Nordahl Petersen, Pernille Andersen, Morten Nielsen and Claus Lundegaard
# BMC Structural Biology 2009, 9:51 doi:10.1186/1472-6807-9-51
#
# Column 1: Class assignment - B for buried or E for Exposed - Threshold: 25% exposure, but not based on RSA
# Column 2: Amino acid
# Column 3: Sequence name
# Column 4: Amino acid number
# Column 5: Relative Surface Accessibility - RSA
# Column 6: Absolute Surface Accessibility
# Column 7: Z-fit score for RSA prediction
# Column 8: Probability for Alpha-Helix
# Column 9: Probability for Beta-strand
# Column 10: Probability for Coil
E A  2WNS.A                    1    0.777  85.680   1.547   0.003   0.003   0.994
E L  2WNS.A                    2    0.281  51.451   0.503   0.802   0.014   0.185
E G  2WNS.A                    3    0.530  41.687   1.558   0.938   0.007   0.055
E P  2WNS.A                    4    0.549  77.846   2.101   0.938   0.007   0.055
B L  2WNS.A                    5    0.074  13.531   0.513   0.938   0.007   0.055
B V  2WNS.A                    6    0.068  10.405   1.064   0.975   0.003   0.022
E T  2WNS.A                    7    0.538  74.662   2.087   0.975   0.003   0.022
E G  2WNS.A                    8    0.296  23.335   2.326   0.970   0.001   0.030
B L  2WNS.A                    9    0.029   5.310   0.164   0.970   0.001   0.030
E Y  2WNS.A                   10    0.326  69.581   1.270   0.923   0.002   0.076
E D  2WNS.A                   11    0.764 110.107   1.486   0.782   0.003   0.216
B V  2WNS.A                   12    0.217  33.307   1.603   0.321   0.003   0.675
E Q  2WNS.A                   13    0.563 100.516   0.817   0.018   0.019   0.964
B A  2WNS.A                   14    0.058   6.348   0.220   0.018   0.088   0.893
B F  2WNS.A                   15    0.088  17.662   0.307   0.022   0.552   0.426
E K  2WNS.A                   16    0.370  76.150   0.997   0.022   0.552   0.426
E F  2WNS.A                   17    0.454  91.178   0.204   0.021   0.451   0.528
E G  2WNS.A                   18    0.387  30.481  -0.578   0.019   0.141   0.840
E D  2WNS.A                   19    0.574  82.742   0.151   0.019   0.141   0.840
B F  2WNS.A                   20    0.126  25.288   0.725   0.021   0.756   0.223
E V  2WNS.A                   21    0.393  60.373   0.761   0.018   0.846   0.136
B L  2WNS.A                   22    0.155  28.289   0.530   0.018   0.846   0.136
```

Figure 2.4. The figure shows an example of the output obtained from NetSurfP.

Table 2.1. Performance measures for secondary structure prediction using CB513.

	Matthews correlation	**% fraction of correct prediction**
H	0.78	84.2
E	0.68	71.9
C	0.65	83.4
Overall	0.72	81.3

The table is listing the prediction performance for the secondary structure prediction method implemented in NetSurfP.

NetSurfP has since its launch in August 2009 become quite popular. With more than 29,500 visits from 113 countries, NetSurfP has made predictions on more than 161,500 sequences corresponding to 65,421,000 amino acids. A total of 41 papers cites or mentions the use of NetSurfP. Since October 15, 2010 NetSurfP has also been available for download for academic users at: http://www.cbs.dtu.dk/cgi-bin/nph-sw_request?netsurfp. Until now 29 individuals have downloaded the software package.

2.4 Paper I

Prelude

The paper entitled *A generic method for assignment of reliability scores applied to solvent accessibility predictions*, published in *BMC Structural Biology*, presents a method for prediction of the surface accessibility of amino acids, along with their secondary structure.

Bent Petersen, Thomas Nordahl Petersen, Pernille Andersen, Morten Nielsen and Claus Lundegaard.
A generic method for assignment of reliability scores applied to solvent accessibility predictions. *BMC Structural Biology*, 9:51, 2009.

The method has been implemented in a web-server, which can be accessed at this address:
http://www.cbs.dtu.dk/services/NetSurfP/

Methodology article

A generic method for assignment of reliability scores applied to solvent accessibility predictions

Bent Petersen[1], Thomas Nordahl Petersen[1], Pernille Andersen[1,2], Morten Nielsen[1] and Claus Lundegaard*[1]

Address: [1]Center for Biological Sequence Analysis - CBS, Department of Systems Biology, Kemitorvet 208, Technical University of Denmark - DTU, DK-2800 Lyngby, Denmark and [2]Centre for Medical Parasitology - CMP, CSS Building 22, University of Copenhagen, DK-1014 Copenhagen, Denmark

Email: Bent Petersen - bent@cbs.dtu.dk; Thomas Nordahl Petersen - tnp@cbs.dtu.dk; Pernille Andersen - pan@cbs.dtu.dk; Morten Nielsen - mniel@cbs.dtu.dk; Claus Lundegaard* - lunde@cbs.dtu.dk

* Corresponding author

Published: 31 July 2009
Received: 10 February 2009
BMC Structural Biology 2009, **9**:51
Accepted: 31 July 2009
doi:10.1186/1472-6807-9-51
This article is available from: http://www.biomedcentral.com/1472-6807/9/51

Abstract

Background: Estimation of the reliability of specific real value predictions is nontrivial and the efficacy of this is often questionable. It is important to know if you can trust a given prediction and therefore the best methods associate a prediction with a reliability score or index. For discrete qualitative predictions, the reliability is conventionally estimated as the difference between output scores of selected classes. Such an approach is not feasible for methods that predict a biological feature as a single real value rather than a classification. As a solution to this challenge, we have implemented a method that predicts the relative surface accessibility of an amino acid and simultaneously predicts the reliability for each prediction, in the form of a Z-score.

Results: An ensemble of artificial neural networks has been trained on a set of experimentally solved protein structures to predict the relative exposure of the amino acids. The method assigns a reliability score to each surface accessibility prediction as an inherent part of the training process. This is in contrast to the most commonly used procedures where reliabilities are obtained by postprocessing the output.

Conclusion: The performance of the neural networks was evaluated on a commonly used set of sequences known as the CB513 set. An overall Pearson's correlation coefficient of 0.72 was obtained, which is comparable to the performance of the currently best public available method, Real-SPINE. Both methods associate a reliability score with the individual predictions. However, our implementation of reliability scores in the form of a Z-score is shown to be the more informative measure for discriminating good predictions from bad ones in the entire range from completely buried to fully exposed amino acids. This is evident when comparing the Pearson's correlation coefficient for the upper 20% of predictions sorted according to reliability. For this subset, values of 0.79 and 0.74 are obtained using our and the compared method, respectively. This tendency is true for any selected subset.

Background

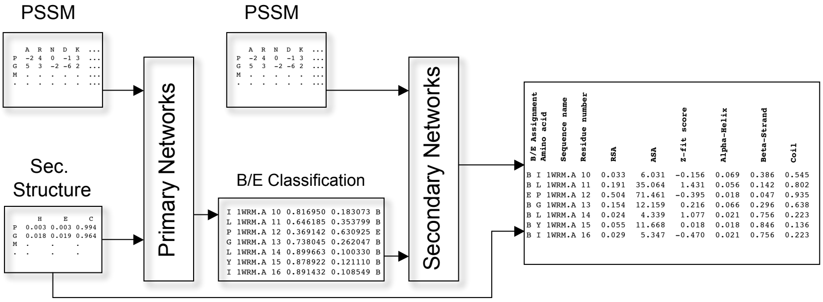
For decades, machine learning has been used as a tool in bioinformatics for predictive purposes. A number of concepts have been implemented in order to estimate the predictive power of the individual methods. The commonly used performance measures have been described in Lundegaard et al. (Lundegaard et al. (2007)). Predictive power is generally estimated from a number of examples that have been excluded from the training process and an overall estimate of the accuracy of the method is calculated. This, however, will not provide information regarding the reliability of each of the individual predictions. For discrete qualitative predictions, the reliability is conventionally estimated as the difference between output scores of selected classes (Rost (1996)). However, many biological problems are quantitative in nature and are therefore more appropriately characterized by a real value than a discrete class. Real value predictions often provide a single output value and the estimation of the accuracy of a given prediction is more complicated than for predictions of discrete classes. Prediction of the solvent accessible surface area (ASA) of amino acid residues within a native folded protein is an example of a real value prediction problem, where the estimation of reliability scores is nontrivial. The ASA for experimentally solved structures is given in Å^2 and the area is calculated by rolling a sphere the size of a water molecule over the protein surface (Connolly (1983)). For comparative and predictive purposes, the ASA is often transformed to a relative surface area (RSA), which is calculated as the ASA of a given amino acid residue in the polypeptide chain, relative to the maximal possible exposure of that residue in the center of a tri-peptide flanked with either glycine (Chothia (1976)) or alanine (Ahmad et al. (2003)). Knowledge of the degree of surface exposure of an amino acid is valuable and it has been used to enhance the understanding of a variety of biological problems including protein-protein interactions (Jones and Thornton (1997a), Jones and Thornton (1997b)), structural epitopes (Haste Andersen et al. (2006)), active sites (Panchenko (2004)), and prediction of disease-related single nucleotide polymorphisms (Mooney (2005)).

Several methods for predicting surface accessibility from the primary protein sequence have been developed often inspired by the related field of protein secondary structure prediction as exemplified with (Pollastri et al. (2002)) implemented in (Cheng et al. (2005)). Generally, the best methods involve the use of advanced machine learning algorithms such as artificial neural networks (ANN) or support vector machines (SVM) combined with evolutionary information (Adamczak et al. (2004), Carugo (2000), Garg et al. (2005), Pollastri et al. (2007), Wang et al. (2007), Xu et al. (2006), Yuan et al. (2002), Yuan and Huang (2004)). The surface accessibility has traditionally been predicted in two classes as either buried or exposed using various more or less arbitrary cut-offs. Recently, real value RSA predictors have been developed thus removing the need to define specific cut-offs (Ahmad et al. (2003)). This change in focus from classifying towards quantitative systems has made it difficult to assess the reliability of a prediction. Previous studies have shown that prediction of the RSA is significantly more accurate for buried compared to exposed amino acids (Dor and Zhou (2007a)). However, the most biologically interesting residues are often exposed, as these are able to interact with the environment. For this reason, it is important to have a good estimate of the reliability, especially for the more exposed amino acid residues. The current best method available for real value surface exposure prediction is Real-SPINE (Dor and Zhou (2007b), Faraggi et al. (2009)). This method exists in a web accessible form, which in addition to the predicted surface accessibility, also provides a score for each prediction that is a measure of the consistency between two predictors (A, B). $RS = 1 - |A - B|$ where A and B are the results from two predictors on solvent accessibility (Dor and Zhou (2007b)). As described this score is solely a consistency score and it has not previously been described to what degree such consistency measures provide information of the reliability of the individual predictions beyond the fact that the most exposed residues are predicted most unreliably.

Here, we have developed a generic method that assigns a reliability score to each surface accessibility prediction as an inherent part of the training process. The method is evaluated on a common set of sequences and compared to other state-of-the art prediction methods. In particular, we investigate to what extent our method for residue-specific reliability prediction is able to discriminate between good and bad predictions in the entire range from completely buried to fully exposed amino acids.

Results

A schematic overview of the NetSurfP method is shown in Figure 2.5. The method consists of two neural network ensembles. The primary networks are



```
   A  R  N  D  K  ...
P  -2 4  0  -1 3  ...
G  5  3  -2 -6 2  ...
M  .  .  .  .  .  ...
.  .  .  .  .  .  ...
```

```
     H     E     C
P  0.003 0.003 0.994
G  0.018 0.019 0.964
M  .     .     .
.  .     .     .
```

```
I 1WRM.A 10 0.816950 0.183073 B
L 1WRM.A 11 0.646185 0.353799 B
P 1WRM.A 12 0.369142 0.630925 E
G 1WRM.A 13 0.738045 0.262047 B
L 1WRM.A 14 0.899663 0.100330 B
Y 1WRM.A 15 0.878922 0.121110 B
I 1WRM.A 16 0.891432 0.108549 B
```

```
B/E Assignment
  Amino acid
       Sequence name
                Residue number
                      RSA       ASA     Z-fit score  Alpha-Helix  Beta-Strand  Coil
B I 1WRM.A 10  0.033    6.031  -0.156  0.069  0.386  0.545
B L 1WRM.A 11  0.191   35.064   1.431  0.056  0.142  0.802
E P 1WRM.A 12  0.504   71.461  -0.395  0.018  0.047  0.935
B G 1WRM.A 13  0.154   12.159   0.216  0.066  0.296  0.638
B L 1WRM.A 14  0.024    4.339   1.077  0.021  0.756  0.223
B Y 1WRM.A 15  0.055   11.668   0.018  0.018  0.846  0.136
B I 1WRM.A 16  0.029    5.347  -0.470  0.021  0.756  0.223
```

Figure 2.5. Graphical overview of the method. Graphic overview of the method used in training of the primary and secondary neural networks. 'PSSM' is a Position-Specific Scoring Matrix. 'Sec. Structure' is the raw output from secondary structure predictions. 'Primary Networks' are an ensemble of artificial neural networks (ANN) and 'B/E Classification' is the raw buried/exposed out-put from these ANNs. 'Secondary Networks' are also an ensemble of ANNs, trained to predict the relative surface exposure of an amino acid. The last box shows output from the web server.

trained on sequence profiles and predicted secondary structure and have two outputs corresponding to buried or exposed, respectively. The higher output defines the predicted category. The secondary networks use these outputs as input together with sequence profiles and have been trained to predict the relative surface exposure of the individual amino acid residues. The proposed reliability prediction method is applied to the secondary networks only.

Primary networks

Classification artificial neural networks (ANNs) were trained to predict whether an amino acid was buried or exposed i.e., below or above 25% of ASA$_{max}$ of the given amino acid. Input to the ANNs was sequence profiles and predicted secondary structures. The prediction performance of the secondary structure prediction in terms of the straight Q3 measure on the CB513 dataset was 81%. Secondary structure predictors were trained to predict H or E classes (see methods), which differs from the CASP classification scheme used by many secondary structure prediction methods (CASP Q3 = 78%).

Using 10-fold cross validation each spanning a series of different network architectures, an ensemble were constructed of the 200 best performing network architectures, determined by the cross validation leave-out test sets (see methods). A test performance of 79.8% accuracy and a Matthews correlation coefficient (MCC) of 0.593 were obtained. This ANN ensemble was also evaluated using the evaluation set CB513. The performance values were 79.0%

correctly classified residues and a MCC of 0.577. These values are compared with the performance obtained by (Dor and Zhou (2007b)) as shown in Table 2.2.

Table 2.2. Evaluated performance for the primary networks.

Method	% Correct	MCC
NetSurfP Classification CB513	79.0	0.577
Dor and Zhou (Dor and Zhou (2007a))	78.8	-

Evaluation of the best performing ANN ensemble using the evaluationset CB513. The columns are the overall %-correct prediction of buried and exposed amino acids and Matthew's correlation coefficient (MCC). Dor and Zhou gives the performance value published by Dor and Zhou (2007b).

Secondary networks

The output classification values from the primary networks were used together with sequence profiles in the form of Position-Specific Scoring Matrices (PSSM) to train the secondary neural networks as also implemented by (Dor and Zhou (2007a)). A significant improvement was obtained compared to bare PSSM input only with respect to linear as well as two-state correlations (data not shown). Several neural network architectures were trained using 10-fold crossvalidation. The best cross-validation leave out test set performance was obtained by using a window size of 11 residues and a number of hidden neurons in the range 25-200. The Real-SPINE method (Dor and Zhou (2007b)) has not previously been evaluated on the CB513 set. We therefore submitted the sequences in the CB513 set to the Real-SPINE 1.0 webserver.

Two sequences were not accepted by the server leaving us with a set of 511 sequences (CB511) used when comparing the performance of NetSurfP and several other methods (Ahmad et al. (2003), Yuan and Huang (2004), Dor and Zhou (2007b), Nguyen and Rajapakse (2006)). The RealSpine and NetSurfP methods perform equally well as shown in Table 2.3.

Prediction and analysis of reliability scores

Neural networks were trained as described in section 'secondary networks'. Real value predictions usually gives one output value between $0-1$ per residue, however, our described method generates two output values for each prediction; the predicted surface accessibility and a reliability of this prediction for each amino acid residue. This was implemented using a modified back-propagation procedure as described in the method section. We evaluated the performance of this method on the CB511 data set and compared the results to those obtained with the method by Dor and Zhou (Dor and Zhou (2007a)). Unless otherwise stated, the performance values were calculated from the RSA. The overall predictive performance of the neural network

Table 2.3. Evaluation of NetSurfP and other surface accessibility predictors

Method	Exposure	Train	CB513/CB511	Method
Ahmad	ASA	-	0.48	ANN
Yuan	ASA	-	0.52	SVR
Nguyan	ASA	-	0.66	Two-Stage SVR
Real-SPINE	ASA	0.74	0.73	ANN
Real-SPINE	RSA	-	0.70	ANN
NetSurfP	ASA	0.75	0.72	ANN
NetSurfP	RSA	0.72	0.70	ANN

Performances are shown for 5 different approaches to predict absolute and relative (RSA) surface accessibility. Methods included in the benchmark are Ahmad: [5], Yuan: (Yuan and Huang (2004)), Nguyen: (Nguyen and Rajapakse (2006)), Real-SPINE: (Dor and Zhou (2007b)), NetSurfP: This work. Train gives the training performance, and CB513/CB511 gives the evaluation performance on the CB513 dataset. Train performance of the Real-SPINE method and evaluation performances for the Ahmad, Yuan, and Nguyen method are taken from the corresponding publications. ANN = Artificial neural networks, SVR = Support vector regression. Pearson's correlation coefficients (PCC) are shown for all methods based on the absolute surface exposure of an amino acid. Also, PCC values are given for relative surface exposure for the two methods NetSurfP and Real-SPINE.

was 0.145 in terms of the mean error, E, and 0.70 in terms of the Pearson's correlation coefficient (PCC), which is similar to the values obtained earlier using the conventional networks (see Table 2.3).

From the network reliability score, we calculated a reliability value as a Z-score as described in methods. Figure 2.6 (left panel) shows the variation in

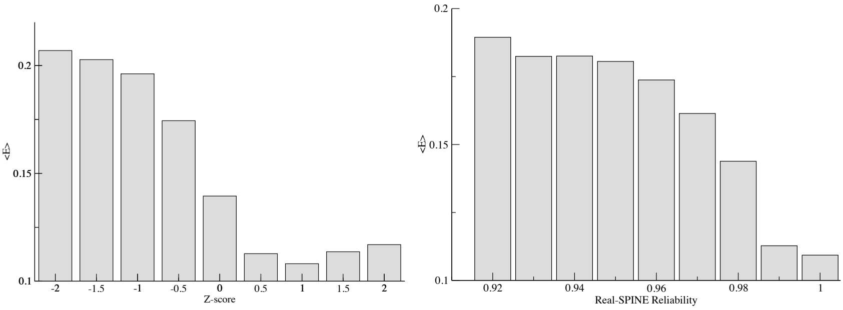


Figure 2.6. The average error as a function of the predicted reliability. The left panel shows NetSurfP Z-score versus mean error, and the right panel shows the consistency reliability score versus mean error.

the mean error as a function of the Z-score reliability from NetSurfP. From this figure, it is apparent that data points with high Z-scores have lower predicted error compared to data points with low Z-scores. We found that the group of data points with positive Z-scores, corresponding to 51% of all data points, achieved a PCC of 0.77, whereas the data points with negative Z-scores achieved a PCC of 0.64. This difference is highly significant ($p < 0.001$, Bootstrap exact estimate).

The Real-SPINE method provides a residue-specific consistency measure associated with each prediction. The relationship between this value and the mean error is shown in the right panel of Figure 2.6. Comparing these two plots suggests that both methods are able to identify the most reliable predictions.

It has previously been reported that amino acid residues, which are predicted to be highly buried tend to have lower predicted error compared to those predicted as exposed (Ahmad et al. (2003), Dor and Zhou (2007b)). To investigate how this might bias the reliabilities we examined the mean predicted error as a function of the predicted exposure when splitting the data in two groups with high (top 50%) and low (bottom 50%) reliability, respectively (Figure 2.7). The plot visualizes how the predictions with a corresponding high Z-score have a lower mean error compared to those with a low Z-score. This is valid for all ranges of predicted exposure. This, on the other hand, is not the case for the consistency scores. Comparing the "high" and "low" reliability groups we see a difference only for residues that were predicted to be buried ($RSA < 0.2$). The same trend is observed when using a cut-off of top 25% and 75% highest predictions for both Real-SPINE and NetSurfP (data not shown).

Likewise, we tested to what degree the two reliability measures are capable of identifying reliable predictions independent of the degree of exposure. The distribution of predicted RSA values for the 25%, 50%, 75% and 80% residues with highest consistency scores was shown for the Real-SPINE (Figure 2.8, left panel) and highest Z-score for NetSurfP (Figure 2.8, right panel), respectively. These figures reveal that the Real-SPINE method predominantly assigns high consistency scores to buried residues, and when filtering out low consistency predictions mostly exposed residues are removed. This can be seen on the insert for Real-Spine (Figure 2.8, left panel) where there is a bias against low RSA. In contrast to this, high NetSurfP Z-score values are found for residues in all exposure ranges. The curve in the insert for NetSurfP (Figure 2.8, right panel), is close to horizontal meaning predictions are equally distributed over the different levels of exposure independent of Z-score reliability threshold. The predictive performance of the 80% residues with highest reliability of the two methods is 0.73 and 0.79 in terms of the PCC for the consistency and the derived Z-score methods, respectively. This difference in predictive performance is highly significant ($p < 0.0001$, Bootstrap exact estimate).

The above results could depend on the chosen cut-off for the fraction of most reliable predictions (80%) that were included in the test. To investigate this bias we took an increasing number of the Z-score/consistency

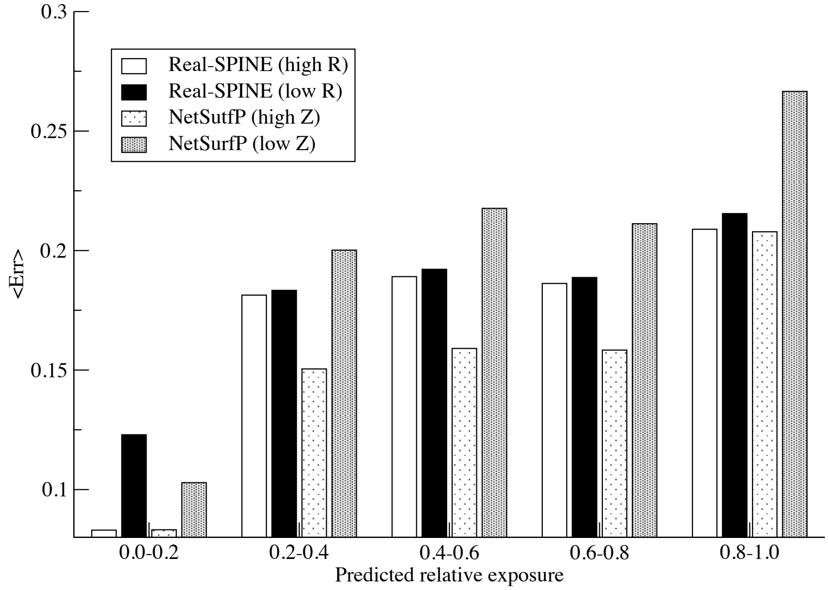


Figure 2.7. Histogram of mean error as a function of predicted exposure values. The bars show the histogram for four groups of predictions with high and low reliabilities: "High R" and "low R" for the consistency method and "high Z" and "low Z" for the NetSurfP method, where "high" is the 50% most reliable predictions according to the chosen reliability score, and "low" is the 50% least reliable predictions.

ranked predictions and calculated the average RSA of the selected sets both regarding predicted and measured RSA. In Table 2.4 it is shown that the predictions from the Real-SPINE with the highest consistency have a strong bias towards buried residues. Using the NetSurfP derived Z-score, no such bias was observed and the ratio between buried/exposed residues was maintained for all levels of reliability, i.e. the mean predicted relative accessibility (P-RSA) equals the mean measured (M-RSA) in each subset. In addition, the PCC of the Z-score filtered NetSurfP predictions is better within nearly all of the most reliable subsets than that of the consistency filtered Real-SPINE predictions, despite the fact that the two methods have close to identical overall performances. Furthermore, the subsets of reliable NetSurfP predictions identified by the Z-score method maintain a constant average of both the predicted surface exposure and the surface exposure calculated from experimentally solved structures independent of the degree of reliability. However, using the consistency filter on Real-SPINE predictions we saw that the average of the predicted or calculated surface exposure decreased (i.e., the relative amount of buried residues increased) as the reliability increases. The final implementation of the NetSurfP method as a web-server

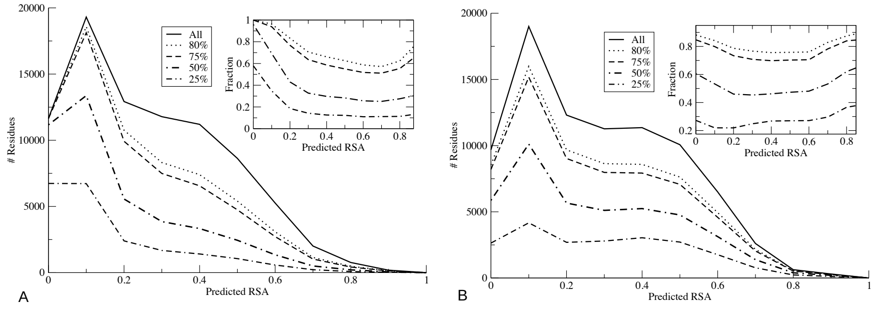


Figure 2.8. Histogram of the number of predicted residues (A: Real-Spine and B: NetSurfP) as a function of the predicted relative exposure value for all residues in the CB511 data set at different cut-offs. The full line shows the calculated (measured) exposure distribution of the full set. The distribution of the 25%, 50%, 75% and 80% most reliably A: Real-Spine predicted residues according to consistency score, and B: NetSurfP predicted residues according to the Z-score, are also shown. Insert shows the number of predicted residues/all predictions in a given threshold as a function of the predicted RSA.

was done by also including the sequences (CB513 set) that were previously only used as an evaluation set. The secondary structure predictor is implemented as part of the NetSurfP web-server. The webserver is available at http://www.cbs.dtu.dk/services/NetSurfP/.

Table 2.4. Evaluation of the Real-SPINE and NetSurfP method on subsets of residues from the CB511 dataset predicted with high reliability

		Real-SPINE				NetSurfP			
%Top	N	RSA	ASA	P-RSA	M-RSA	RSA	ASA	P-RSA	M-RSA
10	8372	0.73	0.74	0.16	0.18	0.77	0.79	0.35	0.35
20	16745	0.73	0.74	0.16	0.18	0.79	0.79	0.31	0.31
25	20931	0.73	0.74	0.17	0.19	0.79	0.79	0.30	0.30
50	41863	0.72	0.74	0.18	0.20	0.77	0.77	0.28	0.28
75	62795	0.71	0.73	0.22	0.24	0.74	0.75	0.28	0.28
80	66981	0.71	0.73	0.23	0.25	0.73	0.74	0.28	0.28
90	75354	0.70	0.73	0.25	0.27	0.72	0.73	0.28	0.28
100	83727	0.70	0.73	0.27	0.29	0.70	0.72	0.29	0.29

%Top and N give the percentage and number of residues selected. RSA and ASA give the Pearson's correlation between predicted and target for relative and absolute surface areas, respectively. P-RSA, and M-RSA give the mean predicted and mean measured RSA values, respectively, on the selected subset of residues.

Discussion

The power of a prediction method is commonly evaluated as an overall estimate of the accuracy of the method in large-scale benchmark experiments. Such evaluation, however, provides no knowledge of the reliability of each of the individual predictions. For discrete, qualitative predictions the reliability is conventionally estimated as the difference between output scores of selected classes. For real value prediction this approach is unfeasible. Here, we have described a new reliability score method, useful for real value predictions. We have designed and implemented the method in a way that assigns reliability scores for each single real value prediction. As an example, the method has been implemented as part of a web-server to predict the relative surface accessible area of amino acids within the three dimensional structure of a protein. By nature, the reliability method is different from other procedures where reliabilities most commonly are obtained by post-processing the output (Rost (1996), Dor and Zhou (2007b)). This method was trained to assign a reliability output to each surface accessibility prediction as an inherent part of the network architecture. This output was then recomputed to a Z-score. In tests to investigate the validity of the calculated Z-score we found that the score could indeed successfully be used to filter out more reliable predictions resulting in a significantly better correlation between predicted and measured values.

The accessible surface area has been found more difficult to predict for exposed than buried amino acids and these findings are still valid (Ahmad et al. (2003), Dor and Zhou (2007a), Dor and Zhou (2007b)). However, we see that NetSurfP Z-scores enable the identification of the most reliable/unreliable predictions for both buried and exposed amino acids. This allows for identification of subsets of highly reliable predictions covering all ranges of surface exposure. This is in contrast to the consistency score, the only other surface accessibility prediction associated reliability method (Dor and Zhou (2007b)), where high reliability scores are predominantly associated with buried amino acids.

The prediction accuracy is compared to Real-SPINE 1.0 (Dor and Zhou (2007b)) as Real-SPINE 1.0 is the server that produces the consistency measures. Furthermore the newly published Real- SPINE 3.0 (Faraggi et al. (2009)) was not available at the time of the evaluation.

Conclusion

In the present context, the developed reliability information is especially valuable when using the surface exposed predictions to estimate other protein structure related features such as fold, B cell epitopes, phosphorylation sites, and active sites. However, the approach is generic and is potentially useful in other types of real value predictions where ANNs have been shown to produce good results.

Materials and Methods

Barton Evaluation dataset, CB513/CB500

The dataset of 513 non-homologous proteins created by Cuff and Barton (Barton (2007), Cuff and Barton (1999)) consists of >84,000 amino acids. It is commonly known as the CB513 dataset. The dataset consist of 117 sequences from the Rost and Sander dataset of 126 non-redundant proteins (Rost and Sander (1993)) and 396 sequences are from the CB396 dataset by Cuff and Barton (Cuff and Barton (1999)). No sequences in the dataset share more than 25% sequence identity. The CB513 dataset was downloaded from the Jpred section at the Barton Group's website http://www.compbio.dundee.ac.uk/∼www-jpred/data/. This dataset is solely used for final evaluations.

Learning/Training dataset, Cull-1764

Protein sequence data was obtained from the RCSB (Research Collaboratory for Structural Bioinformatics) Protein Data Bank (PDB) (Berman et al. (2000)) July 2007 using the protein culling server PISCES (Wang and Dunbrack Jr (2003)) available at http://dunbrack.fccc.edu/PISCES.php. PDB was culled using the following criteria: Maximum sequence percentage identity <= 25%, Resolution <= 2.0 Å, R-factor <= 0.2, Sequence length in the range 30−3,000 amino acids and including full X-ray structures only. This dataset contained 2,263 PDB protein chains, but an additional 197 chains were removed due to parsing errors using the DSSP program (Kabsch and Sander (1983)) and 302 sequences were removed due to more than 25% identity to a sequence within the CB513 set. The final Cull dataset (Cull-1764) is comprised of 1,764 sequences with a total of 417,978 amino acids. Dataset named 'testset' used for optimization of parameters and procedures is always subsets/slices of the Cull-1764 dataset that have been excluded for the particular training session.

Position Specific Scoring Matrices

Sequence profiles as Position-Specific Scoring Matrices (PSSM) were generated for all protein chains in the Cull-1764 and CB513 dataset, using the iterative PsiBLAST program (Altschul et al. (1997)). The query sequences were blasted for four iterations against a local copy of the National Center for Biotechnology Information (NCBI) non-redundant (nr) sequence database, which for speed-up purposes had been homology-reduced to less than 70% sequence identity (Li et al. (2001)). An E-value cut-off of 1×10^{-5} was used.

Relative Solvent Accessibility

The relative solvent accessibility (RSA) is calculated as given by Equation 2.2.

$$RSA = \frac{ASA}{ASA_{MAX}} \cdot 100\% \tag{2.2}$$

RSA is the ratio of the solvent Accessible Surface Area (ASA) of a given residue observed in the three-dimensional structure, over the maximum obtainable solvent exposed area ASA$_{max}$ for the given amino acid residue within an extended tri-peptide flanked with either glycine (Chothia (1976)) or alanine (Ahmad et al. (2003)) residues. Values for the accessible surface area were calculated using the DSSP program (Kabsch and Sander (1983)).

Neural Network Training

Two types of feed-forward neural networks (Rumelhart et al. (1986)) were used in this work: the primary and secondary networks. The primary networks assign one of the classes "Buried" or "Exposed" to each amino acid (see section *Primary Neural Networks*), whereas the secondary networks predict both the real value RSA and the reliability of the prediction in form of a Z-score (see section *Secondary Neural Networks*). A gradient descent method was used to back-propagate the errors and synapses or weights were updated as previously described (Lund et al. (2005)). For the primary networks, amino acids were encoded with both PSSM values and three extra neurons for predicted Helix, Strand and Coil, thus a total of 24 neurons were used to describe an amino acid. The two-class output from the primary networks was subsequently used as input together with PSSM to the secondary neural networks. 10-fold cross-validation was used to train the networks, where 9/10 of the data was used for training and testing was performed on the remaining 1/10, named 'testset'. A graphic overview of the method is shown in Figure 2.5.

Primary Neural Networks

All amino acids in the Cull-1764 dataset were divided into two discrete categories; above and below 25% RSA meaning exposed or buried amino acids, respectively. The RSA values were calculated using the extended gly-X-gly tripeptide state as maximally exposed. In the Cull-1764 dataset the exposed and buried categories comprised 184,757 (44.2%) and 233,221 (55.8%) amino acids, respectively. The primary neural networks were trained using window sizes of 11, 13, 15, 17 and 19, and the following number of hidden units: 10, 20, 25, 30, 40, 50, 75 and 150. This gives a total of 40 different neural network architectures for each of the 10 subsets, giving a total of 400 neural networks. The networks were trained until maximal test set performance with a maximum of 200 epochs, using a learning rate of 0.01. Final ANNs were ranked according to test set performances. Within each of the 10 training/test set groups, we added an increasing number of trained ANNs to a network ensemble from the top of the ranked list until the best test set performance was obtained.

Secondary Neural Networks

Target values, the ratio of ASA and ASA$_{max}$, were assigned for all examples in the Cull-1764 dataset. The ASA$_{max}$ values were calculated using amino acids in an extended ala-X-ala tri-peptide configuration. Amino acids were encoded by use of PSSM scores and two additional values for buried and exposed class predictions obtained from the primary neural networks. A 10-fold cross-validation training was done with window size of 11, and the following number of hidden units: 10, 20, 25, 30, 40, 50, 75, 150 and 200, resulting in a total of 90 neural networks. The best results were obtained using a slow learning rate of 0.005 for a maximum of 300 epochs. For each cross-validation partition, the network architecture that achieved the highest test performance was added to the final ensemble of 10 neural networks.

Implementation of reliability predictions

To derive a method that allows for evaluation of the accuracy of each prediction, a modified feed-forward artificial neural network method was constructed. The method takes the conventional input format defined in terms of a set of input values associated with a given target value. The network produces two output values. One value is the predicted relative surface exposure, and one is a value associated with the reliability of that predicted exposure value. The error function guiding the training of the neural network is shown in Equation 2.3.

$$E = \sum_{i} w_i \left(t_i - o_i\right)^2 + \lambda \left(1 - w_i\right) \tag{2.3}$$

Here, t_i is the target value, o_i is the predicted exposure value, w_i is the predicted reliability and λ is a parameter defining the penalty for introducing low reliability predictions. The optimal value of $\lambda = 0.05$ was determined in a small 5 fold cross-validation benchmark. The rational behind this error-function is that data in the training set that are marginal to the consensus motif will most likely be predicted with the highest error. If this is a systematic error, the network should be able to lower the error by learning the weight value w_i associated with such marginal data. To avoid that all weights are assigned a value of zero, the second penalty term is introduced to balance the loss in error introduced by the weight. This term ensures that only data points that are consistently predicted with large errors are associated with weight values lower than one. The architecture is a conventional three-layer network with one input layer, one hidden layer and one output layer. The network was trained using back-propagation, and the training was stopped when the test error was minimal. Note, that the network is trained using just one target value as input, and produces two output values. Without explicit training values, the network hence learns the predicted reliability intrinsically. It does so by lowering the relative weight on data points with high error.

From the training it became apparent that the two output values (exposure and reliability, respectively) from the network were highly correlated. This is most likely due to the fact that deeply buried residues are relatively simple to predict and hence can be predicted with high reliability in contrast to exposed residues that have more complex characteristics. An example of this correlation is shown in Figure 2.9.

To allow for a direct interpretation of the predicted reliability independent of the predicted exposure value, the predicted reliability values were transformed into Z-scores using the following relation:

$$z_i = \frac{(w_i - w_o\,(e))}{\sigma\,(e)} \tag{2.4}$$

Here, w_o is the reliability baseline value at a predicted exposure value of e, and σ is the baseline-corrected standard deviation at a predicted exposure value of e. The reliability baseline, w_o, and standard deviation, σ, were derived for each test set and network architecture from a fit to the test set predicted values. Test set predictions were grouped into 10 equally populated bins. For each bin, the baseline reliability was estimated from the prediction values in that bin. An example of the Z-score corrected reliability values is shown in Figure 2.9. The final Spearman's rank correlation (Spearman (1904)) between Z-score and error is -0.19.

Secondary Structure Prediction

Secondary structure predictions were generated for all amino acids in the dataset using an artificial neural network-based method described previously (Petersen et al. (2000)). Briefly, the architecture includes combinations of primary networks predicting the three classes Helix, Extended strand or Coil with a secondary network filtering the output predictions from the primary network. For training of the method, a dataset, was downloaded from the PISCES server (Wang and Dunbrack Jr (2003)) on July 10th 2004 and consisted of 2,085 sequences with sequence identity $< 25\%$, Resolution < 2.0 Å, and R-factor < 0.25. The dataset was homology-reduced with respect to the sequences in the CB513 dataset, by use of a Hobohm 1 algorithm (Hobohm et al. (1992)). Sequences in the CB513 dataset were used to evaluate the performance of the secondary structure predictor. Secondary structure in both sets was assigned using DSSP (Kabsch and Sander (1983)) and grouped into 3 classes: The H class comprised by DSSP class H, E class comprised by DSSP class E, and the C class comprised by the remaining DSSP classes; ., G, I, B, S and T. The method was trained using conventional 7-fold cross-validation. The final method was based on a combination of 70 primary and 70 secondary neural networks using input window sizes of 15-23 amino acids, 50 or 75 hidden units.

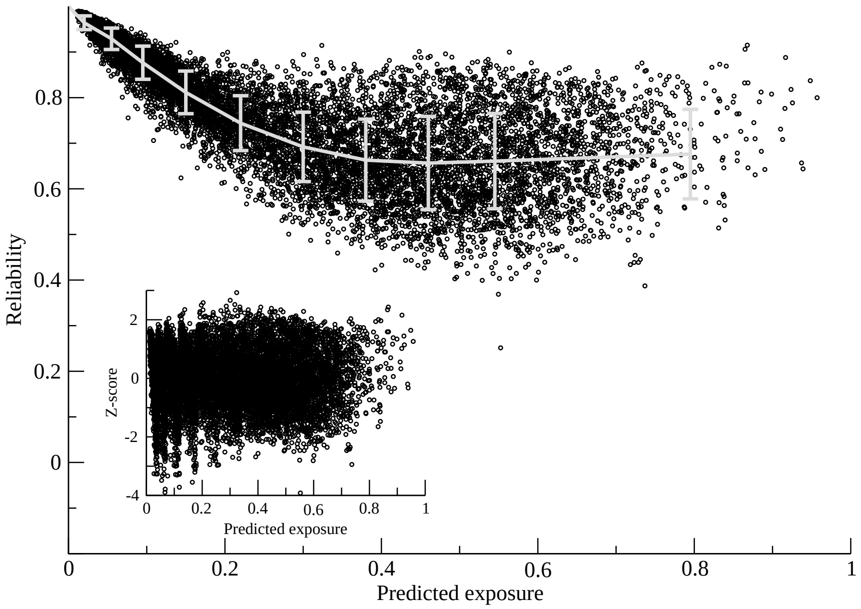


Figure 2.9. Reliability baseline and standard deviation fitting. The reliability is shown as a function of the predicted exposure for the Cull-1764 data set. In grey is shown the fitted reliability baseline and standard deviation. The insert shows the baseline corrected Z-scores as a function of the predicted surface exposure.

Authors' contributions

BP found and curated the third party data used in this work, he performed training, evaluation and selection of ANNs for optimal surface prediction and created the first draft of manuscript. TNP have made substantial contributions to conception and design, assisted with tool and expertise for data curation and ANN development as well as revising the manuscript critically for important intellectual content. PA designed, developed, and described the secondary structure prediction algorithm. MN conceived the idea of ANN error prediction, designed the proper ANN for the task, performed the statistical analysis, and revised the manuscript critically for important intellectual content. CL initiated the development of surface accessibility prediction, and participated in its design and coordination and helped to draft the manuscript. All authors read and approved the final manuscript.

Acknowledgments

We would like to thank Associate professor Thomas Blicher with his help to proofread this manuscript.

This work was funded by the Center for Disease Systems Biology under a grant from the Villum Kann Rasmussen foundation http://www.vkrfondene.dk/, and the EU Commission, BioSapiens (NoE), FP6, contract no.: LSHG-CT-2003-503265.

↺ ↻ ↺

Chapter 3

NetTurnP - in-house tool and webserver

3.1 β-turns

A β-turn is a type of a secondary structure element residing in the coil region. The coil region is often thought of being an unstructured region, but does contain ordered local structures such as α-turns, γ-turns, δ-turns, π-turns and as mentioned also β-turns. β-turns consist of four amino acids (*i*, *i+1*, *i+2* and *i+3*) and they are classified according to their dihedral angles (ϕ and ψ) between amino acids *i+1* and *i+2* (Venkatachalam (1968), Richardson (1981)). Figure 3.1 represents a sequence logo, which illustrates the secondary structure assigned using DSSP (Kabsch and Sander (1983)) of β-turns in the NetTurnP Cull-2220 training dataset. The Cull-2220 training dataset was used for the development of NetTurnP, and it is further described in Chapter 3.3. The plot presented in Figure 3.1 only shows those β-turns that have five or more amino acids before and after the β-turn. The figure illustrates that there is a coil or strand conformation for the two amino acids in front of the β-turn. The other positions, disregarding the four β-turn positions, do not contain a lot of information. The coil or strand conformation in front of the β-turn indicates that the conformation of the protein sequence is going from a hydrophobic strand to a hydrophilic β-turn conformation. This is supported by Figure 3.2 illustrating the buried/exposed information. The maximum information content for the buried/exposed assignment is: $I_c(i) = log_2(1) = 1$. It can be seen that the information content is very low, but nevertheless there is a higher tendency for exposed amino acids at position *i+1* and *i+2* in the β-turn. At position *i* there is a slightly higher tendency for the amino acids to be buried, and at position *i+3* there is no information at all. When the information content of β-turns and α-helices is

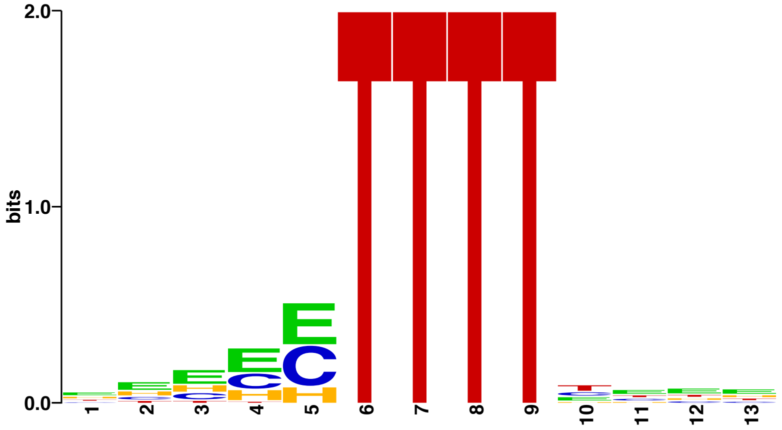


Figure 3.1. Sequence logo showing the information content in the secondary structure of β-turns for the NetTurnP training dataset. The β-turns start at position 6.

compared (see Figure 3.1 and 3.3), it can be noticed that there is an indication of more information available over a longer stretch of amino acids for α-helices than for β-turns. The same tendency can be seen on an amino acid level. Figure 3.4 illustrates the information content for amino acids in β-turns and Figure 3.5 for α-helices. There are more positions with a higher information content in α-helix regions compared to β-turn regions. This indicates that the prediction of β-turns is a more difficult task than the prediction of α-helices. It is also confirmed by the performance obtained for the secondary structure predictor used in this work. The sensitivity for predicting α-helix is 84.2 % with a MCC of 0.78 (unpublished), while the prediction of β-turns has a sensitivity of 75.6 % with a MCC of 0.50. Both methods were evaluated by using the same dataset.

β-turns are defined by both dihedral angle constraints (Venkatachalam (1968), Richardson (1981)) and the restriction that the distance between amino acid *i* and *i+3* has to be smaller than 7 Å and the two central residues *i+1* and *i+2* cannot be helical. β-turns can be further divided into nine different subtypes based on the ϕ and ψ angles between residues *i+1* and *i+2* (Venkatachalam (1968), Richardson (1981)). The standard nomenclature for the β-turn types are: I, I', II, II', VIII, VIa1, VIa2, VIb and IV (Hutchinson and Thornton (1994). The dihedral angles used in the program PROMOTIF (Hutchinson and Thornton (1996)) can be seen in Supplementary Table S7. PROMOTIF is a program which takes a PDB file (Berman et al. (2000)) as input, and then calculates the different β-turn types based on the angles given in the PDB file.

Occassionally β-turns are stabilized with a hydrogen bond between the N-H of residue *i* and the C=O of residue *i+3*. In cases where no hydrogen bond

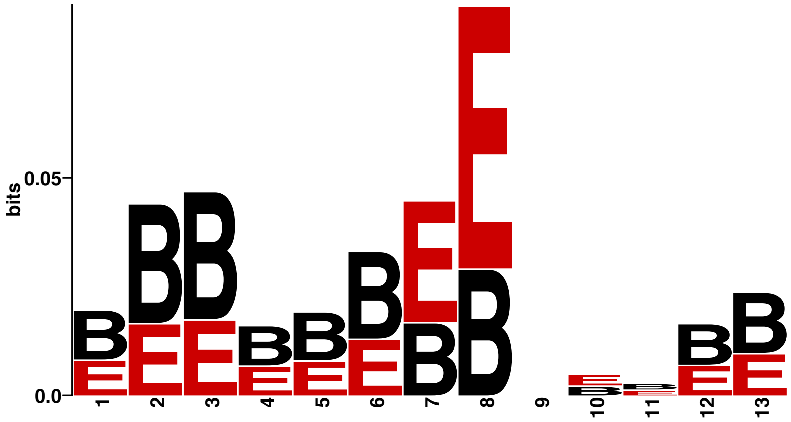


Figure 3.2. Sequence logo showing the information content for the exposure of the amino acids located around the β-turns in the NetTurnP training dataset. The β-turns start at position 6.

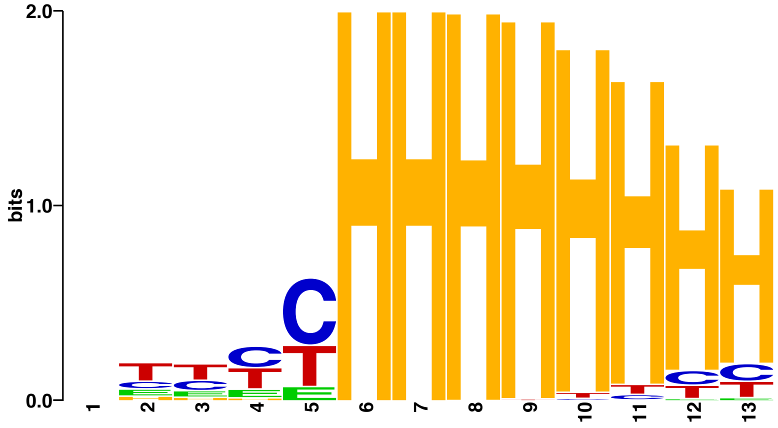


Figure 3.3. Sequence logo showing the information content in the secondary structure area for α-helixes in the NetTurnP training dataset. The α-helices start at position 6.

is found, the β-turn is referred to as an open β-turn (Fuchs and Alix (2005)). β-turns are very interesting in the way that they are the most abundant type of turn structures, which can be found in protein structures. Approximately 25% of all amino acids in protein structures are located in a β-turn and about 58% of all β-turns are composed of different overlapping β-turn types (Hutchinson and Thornton (1994)). Analysing the human proteome release 25H, NetTurnP predicted 10,652,309 amino acids out of 30,407,816 amino acids (35.03%) to be located in a β-turn region. It is approximately 10% more than what was found in the literature.

β-turns play an important role in the formation of compact shapes in proteins and are often referred to as orienting structures due to the fact that they are able to reverse the direction of a protein chain. Besides being very abundant they are also often accessible and generally hydrophilic, two characteristics of antigenic regions (Rose et al. (1985)). For this reason β-turns are suitable candidates for being involved in molecular recognition processes. β-turn types I and II have also been found to be important for SH2 domains (Ettmayer et al. (1999)).

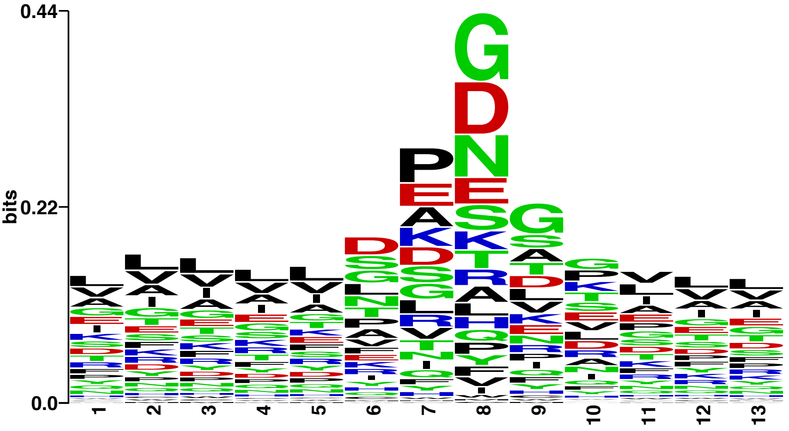


Figure 3.4. Sequence logo illustrating the amino acid preference for β-turns in the NetTurnP training dataset. The β-turns start at position 6.

Analysing the Cull-2220 training dataset consisting of 2220 protein sequences, it was found that the most frequently observed amino acids in β-turns compared to the amino acid at any position were: **Gly** (11.6%/7.2%), **Asp** (8.9%/5.9%), **Ser** (7.1%/6.1%), **Pro** (7.0%/4.6%), **Ala** (6.4%/7.8%), **Asn** (6.3%/4.2%) and **Glu** (6.3%/7.0%). These amino acid residues are hydrophilic or small, where Pro is special due to its fixed and rigid structure making it suitable to reverse the direction of a protein chain. It can be seen that Gly, Asp, Ser, Pro and Asn occur more often in β-turns than in general, and that Ala and Glu occur less frequently. A full table listing all amino acids

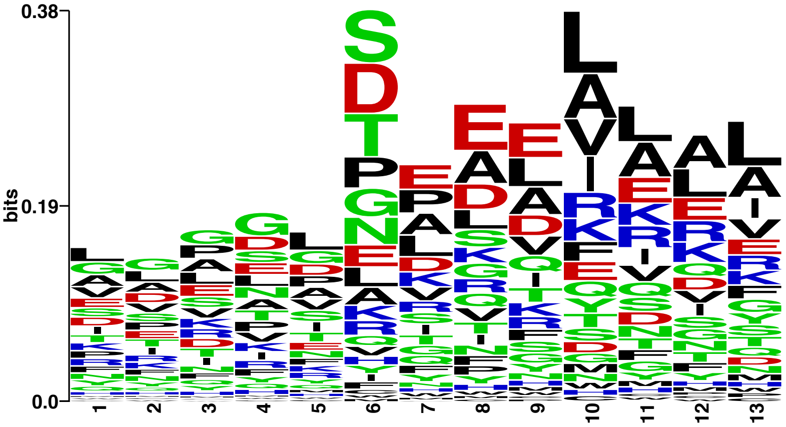


Figure 3.5. Sequence logo illustrating the amino acid preference for α-helices in the NetTurnP training dataset. The α-helices start at position 6.

can be seen in Supplementary Table S6. Figure 3.4 illustrates a sequence logo for the β-turns in the Cull-2220 training dataset.

As mentioned β-turns are suitable candidates for being involved in molecular recognition processes due to their hydrophilic properties and the fact that they are often solvent accessible. Pellequer et. al. (Pellequer et al. (1991)) found that 50% of the linear B-cell epitopes in a small dataset of 11 proteins were located in turn regions, and both β-turn and coil formations have also previously been used to predict linear epitopes (Alix (1999)).

To analyse the frequency of β-turns in discontinuous B-cell epitopes, a dataset with 75 experimentally determined antigen-antibody structures with assigned epitope residues was downloaded from the supplementary section of DiscoTope (Haste Andersen et al. (2006)). A discontinuous B-cell epitope is a conformational epitope, which is composed of discontinuous regions of the antigens amino acid sequence. In the 3D structure these epitopes fold to a conformation, which is able to interact with the paratope of the antibody. It was found that there is an overrepresentation with a factor of 2 (data not shown) of β-turns in discontinuous B-cell epitopes, which correlates well with the findings by Pellequers and co-workers. It indicates that the prediction of β-turns can further improve immunological feature predictions.

In order to predict whether or not an amino acid is located in a β-turn and to predict the presence of the nine β-turn types, NetTurnP was developed. NetTurnP achieves a Q_{total} of 78.2% with a MCC of 0.50 and an AUC of 0.864 for the β-turn/not-β-turn classification problem. All β-turn type performances can be seen in Table 3.1. From the table it can also be seen that some of the β-turn types have a fairly low performance. This is most likely due to the scarce number of examples available for the training of the networks. Some β-turn types contain very few examples. Type VIb (MCC = 0.11), VIa1 (MCC = 0.07) and VIa2 (MCC = 0.03) only have 297, 183 and 69 angles respectively, which is a very small amount. It could have been interesting to redefine the angles defining each β-turn type, so there would be less examples occurring in the type IV, which is the category for all the β-turns, which do not fit in any other type. It would also include more examples in the already defined β-turn types. In the NetTurnP dataset 16,478 angles are assigned to type IV, and many of these could have been assigned to other types if the angles were redefined.

Table 3.1. Performance measures for all β-turn types based on evaluation using BT426.

Prediction method	**Q_{total}**	**PPV**	**Sens**	**Spec**	**MCC**	**AUC**
Type I	78.8	28.1	74.6	79.2	0.36	0.860
Type I'	89.3	8.4	74.8	89.5	0.23	0.917
Type II	85.8	17.1	71.5	86.4	0.31	0.893
Type II'	87.5	4.1	75.5	87.6	0.16	0.907
Type IV	71.0	21.0	71.5	71.0	0.27	0.792
Type VIII	70.2	6.8	75.0	70.1	0.16	0.806
Type VIb	87.1	1.9	83.1	87.1	0.11	0.937
Type VIa1	89.0	1.0	61.9	89.0	0.07	0.874
Type VIa2	89.4	0.3	48.5	89.4	0.03	0.835

The table is listing results for all β-turn types based on evaluation using the BT426 evaluation dataset (see chapter 3.3) on the type specific networks.

3.2 NetTurnP in-house tool and web-server

NEtTurnP is both available as an in-house tool and as a web-server. NetTurnP can be accessed from http://www.cbs.dtu.dk/services/NetTurnP/, where the user has options presented in Figure 3.8. In this example a protein sequence with the name 2WNS.A.1 is submitted, and the user has selected to receive a prediction for the β-turn type I and II. Output from the server is shown in Figure 3.9.

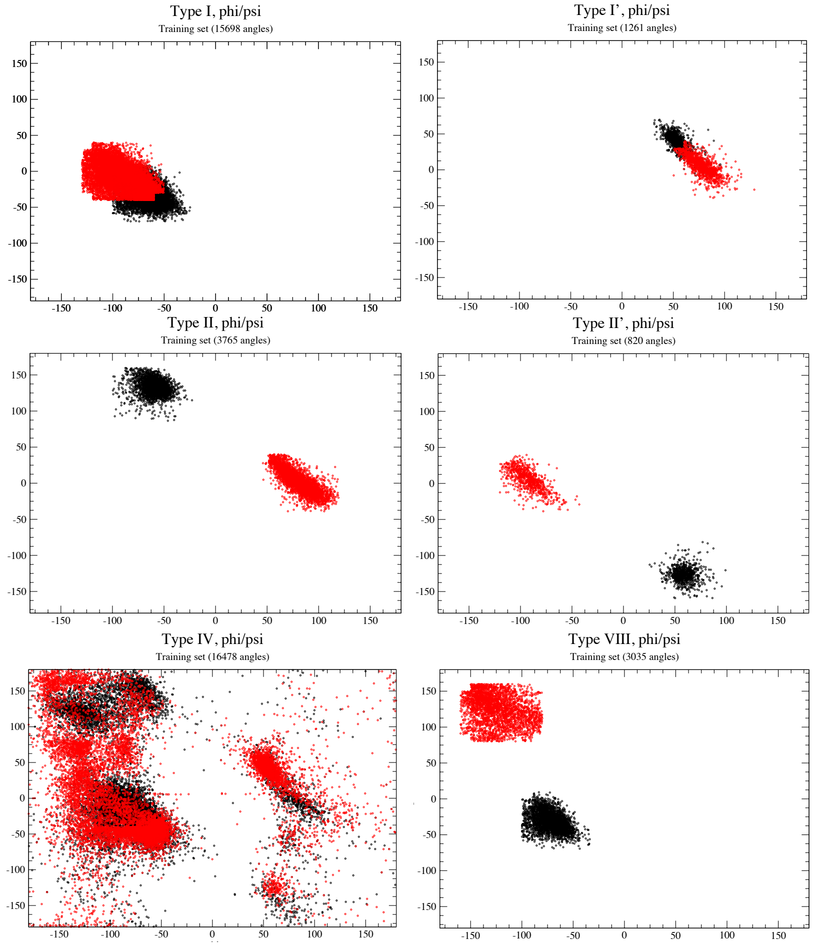


Figure 3.6. The figures illustrate the psi (X-axis) and phi (Y-axis) angles for the β-turn types I, I', II, II', IV and VIII. Black dots corresponds to amino acid *i+1* and red to amino acid *i+2*.

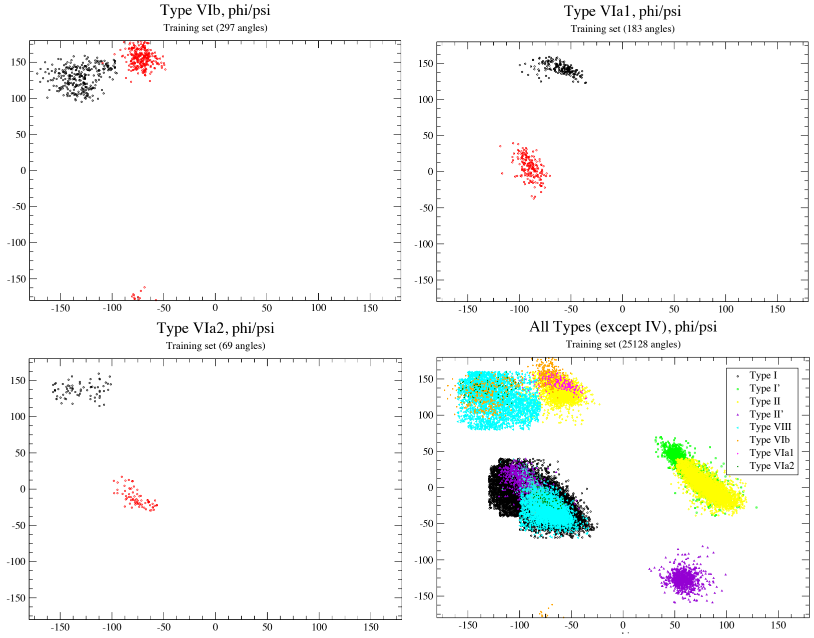
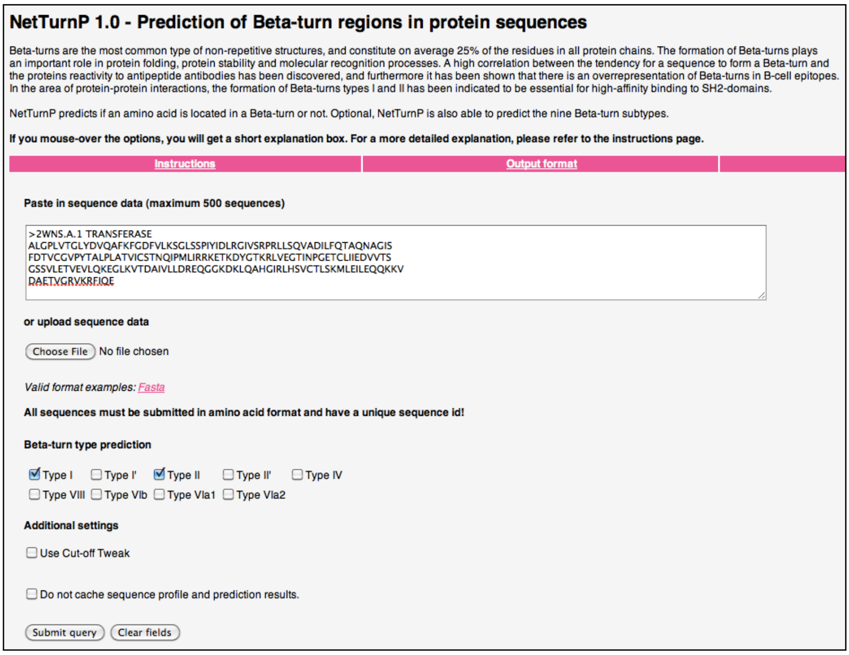


Figure 3.7. The figures illustrate the psi (X-axis) and phi (Y-axis) angles for the β-turn types, VIb, VIa1, VIa2. Black dots corresponds to amino acid *i+1* and red to amino acid *i+2*. The last figure shows all of the β-turn types at the same plot (except type IV).



NetTurnP 1.0 - Prediction of Beta-turn regions in protein sequences

Beta-turns are the most common type of non-repetitive structures, and constitute on average 25% of the residues in all protein chains. The formation of Beta-turns plays an important role in protein folding, protein stability and molecular recognition processes. A high correlation between the tendency for a sequence to form a Beta-turn and the proteins reactivity to antipeptide antibodies has been discovered, and furthermore it has been shown that there is an overrepresentation of Beta-turns in B-cell epitopes. In the area of protein-protein interactions, the formation of Beta-turns types I and II has been indicated to be essential for high-affinity binding to SH2-domains.

NetTurnP predicts if an amino acid is located in a Beta-turn or not. Optional, NetTurnP is also able to predict the nine Beta-turn subtypes.

If you mouse-over the options, you will get a short explanation box. For a more detailed explanation, please refer to the instructions page.

Paste in sequence data (maximum 500 sequences)

```
>2WNS.A.1 TRANSFERASE
ALGPLVTGLYDVQAFKFGDFVLKSGLSSPIYIDLRGIVSRPRLLSQVADILFQTAQNAGIS
FDTVCGVPYTALPLATVICSTNQIPMLIRRKETKDYGTKRLVEGTINPGETCLIIEDVVTS
GSSVLETVEVLQKEGLKVTDAIVLLDREQGGKDKLQAHGIRLHSVCTLSKMLEILEQQKKV
DAETVGRVKRFIQE
```

or upload sequence data

No file chosen

Valid format examples: Fasta

All sequences must be submitted in amino acid format and have a unique sequence id!

Beta-turn type prediction

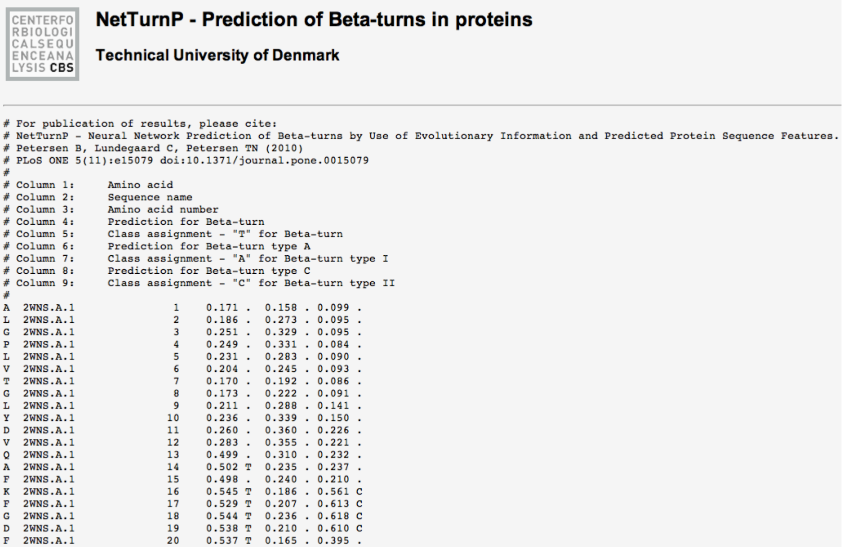
☑ Type I ☐ Type I' ☑ Type II ☐ Type II' ☐ Type IV
☐ Type VIII ☐ Type VIb ☐ Type VIa1 ☐ Type VIa2

Additional settings

☐ Use Cut-off Tweak

☐ Do not cache sequence profile and prediction results.

Figure 3.8. The figure shows the NetTurnP web-server protein sequence submission form.



NetTurnP - Prediction of Beta-turns in proteins

Technical University of Denmark

```
# For publication of results, please cite:
# NetTurnP - Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features.
# Petersen B, Lundegaard C, Petersen TN (2010)
# PLoS ONE 5(11):e15079 doi:10.1371/journal.pone.0015079
#
# Column 1:     Amino acid
# Column 2:     Sequence name
# Column 3:     Amino acid number
# Column 4:     Prediction for Beta-turn
# Column 5:     Class assignment - "T" for Beta-turn
# Column 6:     Prediction for Beta-turn type A
# Column 7:     Class assignment - "A" for Beta-turn type I
# Column 8:     Prediction for Beta-turn type C
# Column 9:     Class assignment - "C" for Beta-turn type II
#
A  2WNS.A.1              1    0.171 .  0.158 . 0.099 .
L  2WNS.A.1              2    0.186 .  0.273 . 0.095 .
G  2WNS.A.1              3    0.251 .  0.329 . 0.095 .
P  2WNS.A.1              4    0.249 .  0.331 . 0.084 .
L  2WNS.A.1              5    0.231 .  0.283 . 0.090 .
V  2WNS.A.1              6    0.204 .  0.245 . 0.093 .
T  2WNS.A.1              7    0.170 .  0.192 . 0.086 .
G  2WNS.A.1              8    0.173 .  0.222 . 0.091 .
L  2WNS.A.1              9    0.211 .  0.288 . 0.141 .
Y  2WNS.A.1             10    0.236 .  0.339 . 0.150 .
D  2WNS.A.1             11    0.260 .  0.360 . 0.226 .
V  2WNS.A.1             12    0.283 .  0.355 . 0.221 .
Q  2WNS.A.1             13    0.499 .  0.310 . 0.232 .
A  2WNS.A.1             14    0.502 T  0.235 . 0.237 .
F  2WNS.A.1             15    0.498 .  0.240 . 0.210 .
K  2WNS.A.1             16    0.545 T  0.186 . 0.561 C
F  2WNS.A.1             17    0.529 T  0.207 . 0.613 C
G  2WNS.A.1             18    0.544 T  0.236 . 0.618 C
D  2WNS.A.1             19    0.538 T  0.210 . 0.610 C
F  2WNS.A.1             20    0.537 T  0.165 . 0.395 .
```

Figure 3.9. The figure shows an example of the output obtained from NetTurnP.

3.3 Paper II

Prelude

The paper entitled *NetTurnP – Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features*, published in *PLoS ONE*, presents a method for prediction of β-turns and the nine β-turn types.

Bent Petersen, Claus Lundegaard and Thomas Nordahl Petersen. NetTurnP – Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features. *PLoS ONE*, 5(11):e15079, 2010.

The method has been implemented in a web-server, which can be accessed at this address: http://www.cbs.dtu.dk/services/NetTurnP/

Methodology article

NetTurnP – Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features

Bent Petersen [1], Claus Lundegaard [1], and Thomas Nordahl Petersen *[1]

[1]Department of Systems Biology, Center for Biological Sequence Analysis (CBS), Technical University of Denmark, Lyngby, Denmark * Email: tnp@cbs.dtu.dk;

Citation: Petersen B, Lundegaard C, Petersen TN (2010) NetTurnP – Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features. PLoS ONE 5(11): e15079. doi:10.1371/journal.pone.0015079
Editor: Andreas Hofmann, Griffith University, Australia

Received Aug 2, 2010; **Accepted** Oct 19, 2010; **Published** Nov 30, 2010

Funding: These authors have no support or funding to report.
Competing Interests: The authors have declared that no competing interests exist.

Abstract

β-turns are the most common type of non-repetitive structures, and constitute on average 25% of the amino acids in proteins. The formation of β-turns plays an important role in protein folding, protein stability and molecular recognition processes. In this work we present the neural network method NetTurnP, for prediction of two-class β-turns and prediction of the individual β-turn types, by use of evolutionary information and predicted protein sequence features. It has been evaluated against a commonly used dataset BT426, and achieves a Matthews correlation coefficient of 0.50, which is the highest reported performance on a two-class prediction of β-turn and not-β-turn. Furthermore NetTurnP shows improved performance on some of the specific β-turn types. In the present work, neural network methods have been trained to predict β-turn or not and individual β-turn types from the primary amino acid sequence. The individual β-turn types I, I', II, II', VIII, VIa1, VIa2, VIba and IV have been predicted based on classifications by PROMOTIF, and the two-class prediction of β-turn or not is a superset comprised of all β-turn types. The performance is evaluated using a golden set of non-homologous sequences known as BT426. Our two-class prediction method achieves a performance of: MCC = 0.50, $Q_{total} = 82.1\%$, sensitivity = 75.6%, PPV = 68.8% and AUC = 0.864. We have compared our performance to eleven other prediction methods that obtain Matthews correlation coefficients in the range of 0.17−0.47. For the type specific β-turn predictions, only type I and II can be predicted with reasonable Matthews correlation coefficients, where we obtain performance values of 0.36 and 0.31, respectively.

Conclusion: The NetTurnP method has been implemented as a webserver, which is freely available at http://www.cbs.dtu.dk/services/NetTurnP/. NetTurnP is the only available webserver that allows submission of multiple sequences.

Introduction

The secondary structure of a protein can be classified as local structural elements of α-helices, β-strands and coil regions. The latter is often thought of as unstructured regions, but do contain ordered local structures such as α-turns, γ-turns, δ-turns, π-turns, β-turns, bulges and random coil structures (Rose et al. (1985), James and Poet (1987)). Turns are defined by a distance that is less than 7 Å between Cα-atoms *i*, *i+2* for γ-turns, *i*, *i+3* for β-turns, *i*, *i+4* for α-turns and *i*, *i+5* for π-turns. Within each turn class, a further classification can be made based on the backbone dihedral angles phi and psi.

β-turn types are classified according to the dihedral angles (ϕ and ψ) between amino acid residues *i+1* and *i+2* (Venkatachalam (1968), Richardson (1981)). The standard nomenclature for the β-turn types are: I, I', II, II', VIII, VIa1, VIa2, VIb and IV (Hutchinson and Thornton (1994)). The dihedral angles for the 9 turn types are shown in Supplementary Table S7.

A β-turn thus involves four amino acid residues, where the two central residues, *i+1* and *i+2*, cannot be helical. Occasionally β-turns are stabilized with a hydrogen bond between the N-H of residue *i* and the C = O of residue *i+3*. In cases where no hydrogen bond is found, the β-turn is referred to as an open β-turn (Fuchs and Alix (2005)).

β-turns are the most abundant type of turn structure found in proteins. They play an important role in the formation of compact shapes in proteins, and are often referred to as orienting structures due to the fact that they have the ability to reverse the direction of a protein chain. Approximately 25% of amino acids in protein structures are located in a β-turn and about 58% of all β-turns are composed of different overlapping β-turn types (Hutchinson and Thornton (1994)).

Prediction of β-turns started in the 1970s where the first β-turn prediction methods relied on statistical information derived from three-dimensional protein structures (Rose et al. (1985), Hutchinson and Thornton (1994), Chou and Fasman (1974), Chou and Fasman (1979), Garnier and Robson (1989), Garnier et al. (1978)). The method implemented by Zhang and Chou (Zhang and Chou (1997)) considered the pairing of the first and the fourth residue, and of the second and the third residue in a β-turn, and the predictive performance reached a Matthews correlation coefficient of 0.17. The work by Fuchs and Alix (Fuchs and Alix (2005)) used statistical methods combined with information obtained from regular secondary structure prediction. Combined with propensity scores and use of evolutionary information, they achieved a Matthews correlation coefficient of 0.41.

The most accurate β-turn predictors today utilize machine-learning methods, although the first approaches did not reach the performance obtained by the best statistical methods. The first method that predicted β-turns by use of neural networks was implemented by McGregor et al. (McGregor et al. (1989)) achieving a Matthews correlation coefficient of 0.20. Ten years later Shepherd et al. (Shepherd et al. (1999)) added secondary structure predictions and the use of a two-layered network architecture (BTPRED method) and obtained a Matthews correlation coefficient of 0.35. Using a k-nearest-neighbor approach, a method by Kim (Kim (2004)) reached a Matthews correlation coefficient of 0.40. Kaur et al. (Kaur and Raghava (2002), Kaur and Raghava (2004)) further enhanced the performance of β-turn prediction by use of secondary structure predictions and evolutionary information in form of position specific scoring matrices as input to the neural networks (BetaTPred2 method) (Kaur and Raghava (2003)). Using a uniform dataset of 426 non-homologues proteins (BT426) they obtained a Matthews correlation coefficient of 0.43. Recently support vector machines have become more widely used in the field of β-turn prediction, which is seen by the work of Zhang et al. (Zhang et al. (2005)) and Liu et al. (E-SSpred method) (Liu et al. (2009)). Using support vector machines with multiple alignments and secondary structure predictions from PSIPRED (McGuffin et al. (2000)), Zhang et al. obtained a Matthews correlation coefficient of 0.45, which was slightly higher than the E-SSpred method. ESSpred reached a Matthews correlation coefficient of 0.44, but they were the first to break the 80% accuracy (Q_{total}) barrier and achieved a Q_{total} of 80.9%, compared to 77.3% by Zhang et al.

Zheng and Kurgan (Zheng and Kurgan (2008)) applied support vector machines using a feature space consisting of position specific scoring matrices and secondary structure predictions from four different methods. After feature reduction, using 90 features, they obtained a Matthews correlation coefficient of 0.47. A similar performance was reached by Hu and Li (Hu and Li (2008)) with a method based on support vector machines using features from position conservation scoring functions. Their method obtained a Matthews correlation coefficient of 0.47 using 7-fold cross-validation on the BT426 dataset.

β-turns are often accessible and generally hydrophilic, two characteristics of antigenic regions (Rose et al. (1985)). For this reason they are suitable candidates for being involved in molecular recognition processes. Pellequer et al. (Pellequer et al. (1991)) found that 50% of the linear B-cell epitopes in a small dataset of 11 proteins were located in turn regions. Thus prediction of β-turns could improve the prediction of epitopes. Krchnak et al. (Krchnak et al. (1987)) found that the parts of a protein, which can induce protein-reactive anti-peptide anti-bodies, mostly reside in regions that have a high tendency to form β-turns. A more recent article by the same authors showed that peptide sequences including a β-turn conformation tended to induce antibodies that were able to cross-react with the parent protein (Krchnak et al. (1989)). β-turn and coil conformations has also previously been used to predict linear epitopes (Alix (1999)). Furthermore, β-turn types I and II, are important for binding between phospho-peptides and SH2-domains (Ettmayer et al. (1999)).

NetTurnP is a new method trained to predict β-turns and the corresponding β-turn type using two layers of neural networks. An improved performance is shown compared to other prediction methods. It has been implemented as a webserver, which is freely accessible at http://www.cbs.dtu.dk/services/NetTurnP/.

Results

Neural network setup

A schematic overview of the final NetTurnP method is shown in Figure 3.10. The method consists of two artificial neural network layers. Several second layer network setups were tested in order to find the architecture with the highest cross-validated MCC value based on training set sequences. These different setups can be seen schematically in Supplementary Table S3. The setups gave similar performances as seen in Figure 3.11, however, we chose the best setup (M) for the final NetTurnP method.

First layer networks

Classification artificial neural networks, β-turn-G, were trained to predict whether or not an amino acid was located in a β-turn. Input to the networks was sequence profiles in form of PSSM's, predicted secondary structure and surface accessibility. Using 10-fold cross validation spanning a series of different network architectures, an ensemble was constructed of the best 100 network architectures, determined by cross validation leave-out tests (see methods). A cross-validated test performance of Q_{total} =77.8%, PPV =51.3%, Sens = 73.1%, MCC = 0.47 and an AUC of 0.846 was obtained.

Furthermore, position specific networks, β-turn-P as described in materials and methods, were also trained in order to increase the predictive performance of the second level networks. Test performances for these networks can be seen in Supplementary Table S4.

Second layer networks

The output from the first layer networks was used as an input to the second layer networks. The final method uses predictions from the β-turn-P

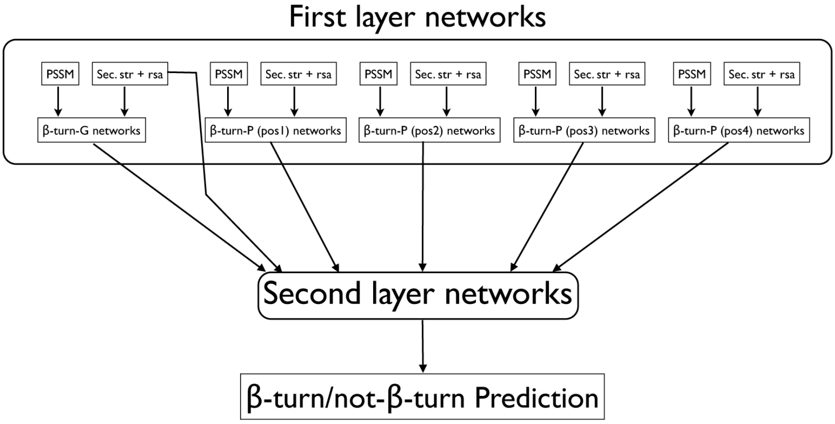


Figure 3.10. Graphical overview of the method used in training of the first and second layer networks. 'PSSM' is a Position-Specific Scoring Matrix. 'Sec. str + rsa' is secondary structure and surface accessibility predictions obtained from NetSurfP (Petersen et al. (2009)). Networks with the abbreviation 'pos' refer to networks that predict specific positions in a β-turn. First layer networks are all ensembles of artificial neural networks where output was used for training in the second layer networks. doi:10.1371/journal.pone.0015079.g001

and β-turn-G networks, including secondary structure and relative surface accessibility predictions from NetSurfP (Petersen et al. (2009)). An ensemble of 10 network architectures was selected corresponding to the top ranking network architecture within each of the subsets, based on the leave-out performance. Further increasing the number of architectures in the ensemble did not increase the performance (Supplementary Figure S1). A cross-validated test performance of Q_{total} = 78.8%, PPV = 53.0%, sensitivity = 71.5% and an MCC of 0.48 with an AUC of 0.849 was obtained. Results for both the first and second layer network test performances are shown in Supplementary Table S5. All performances increased from the first to the second layer networks, except for the sensitivity, which decreased 1.6 percentage points.

The neural network ensemble was also evaluated against the BT426 dataset. The performance values achieved were: Q_{total} = 78.2%, PPV = 54.4%, sensitivity = 75.6% and a MCC of 0.50 with an AUC of 0.864. The ROC curve for the evaluation of the NetTurnP is shown in Figure 3.12. A 7-fold cross validation performed on the BT426 dataset showed that the result obtained is very comparable to the general NetTurnP method as can be seen in Table 3.2.

The Q_{total} measure can be optimized, but at the expense of a lower MCC and sensitivity. We analyzed this relationship by varying the cut-off for a positive prediction as seen in Figure 3.13. A cut-off of 0.61 gave the highest

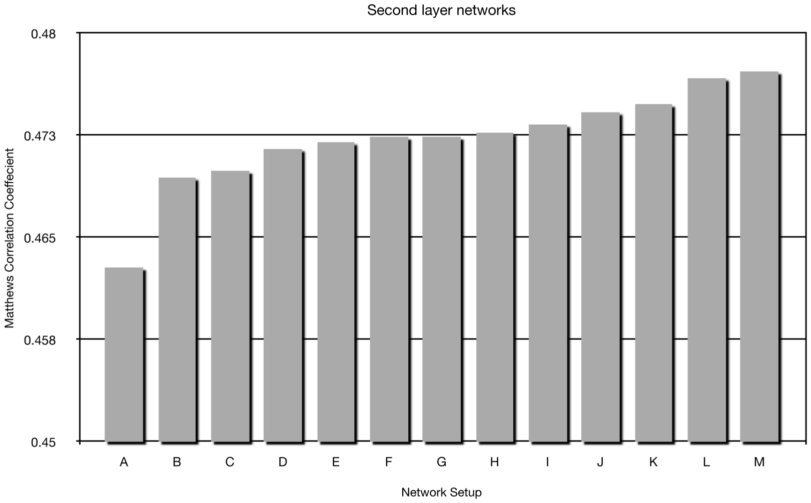


Figure 3.11. Test MCC performance on the Cull-2220 dataset, for different setups of the second level network. The performance is the average from an ensemble of 10 network architectures for each setup. Abbreviations for the setups are as follows: β-turn-P = position specific first layer predictions, β-turn-G = general β-turn/not-β-turn first layer predictions, sec-rsa = secondary structure and surface accessibility predictions from NetSurfP (Petersen et al. (2009)), PSSM = Position Specific Scoring Matrices. The setups are composed as follows: A = PSSM + sec-rsa, B = PSSM + β-turn-G + sec-rsa, C =PSSM + β-turn-G, D= PSSM + β-turn-P, E = β-turn-P, F = β-turn-G + sec-rsa, G= β-turn-G, H= PSSM + β-turn-P + sec-rsa, I = β-turn-P + sec-rsa, J =PSSM + β-turn-P + β-turn-G + sec-rsa, K = PSSM + β-turn-P + β-turn-G, L = β-turn-P + β-turn-G, M= β-turn-P + β-turn-G + sec-rsa. doi:10.1371/journal.pone.0015079.g002

Q_{total} of 82.5% and MCC of 0.46 on the test set, whereas using our default cut-off (0.50) gave a Q_{total} of 78.8 and an MCC of 0.48.

Using this cut-off of 0.61 for the evaluation dataset resulted in a Q_{total} of 82.1% and an MCC of 0.48 as shown in Table 3.2.

Predicted and assigned β-turns are illustrated on the PDB structure 2WNS:A in Figure 3.14. It is a transferase with 197 amino acids where 31 amino acids were assigned by PROMOTIF as being located in a β-turn. Prediction of β-turns was done using the NetTurnP and NetTurnP-tweak methods to show the effect of a tweaked Q_{total} performance. The performance using NetTurnP on 2WNS:A gave Q_{total} = 87.3%, PPV = 55.8%, sensitivity = 93.6% with a MCC of 0.66 and an AUC of 0.955. Using NetTurnP-tweak the protein chain was predicted to a precision of Q_{total} =89.3%, PPV = 100%, sensitivity = 32.3% with a MCC of 0.54. The AUC value was unchanged.

Table 3.2. Comparison of NetTurnP with other β-turn prediction methods

Prediction method	Q_{total}	PPV	Sens	Spec	MCC	AUC
NetTurnP	78.2	54.4	75.6	79.1	0.50	0.864
NetTurnP-tweak	82.1	68.8	50.9	92.4	0.48	0.864
NetTurnP BT426 7-fold	78.1	54.4	74.2	79.5	0.49	0.853
DEBT	79.2	54.8	70.1	N/A	0.48	0.84
E-SSpred	80.9	63.6	49.2	N/A	0.44	0.84
BTNpred	80.9	62.7	55.6	N/A	0.47	N/A
SVM	79.8	55.6	68.9	N/A	0.47	0.87
MOLEBRNN	77.9	53.9	66.0	N/A	0.45	0.832
BTSVM	78.7	56.0	62.0	N/A	0.45	N/A
BetaTPred2	75.5	49.8	72.3	N/A	0.43	0.77
COUDES	75.5	49.8	66.6	N/A	0.41	N/A
KNN	75.0	46.5	66.7	N/A	0.40	N/A
BTPRED	74.9	55.3	48.0	N/A	0.35	N/A
1-4 and 2-3 correlation model	59.1	32.4	61.9	N/A	0.17	N/A

Results are based on the BT426 evaluation dataset. All performance measures have been described in the methods section. NetTurnP is referring to the final performance after the second layer networks, NetTurnP-tweak is the approach that was tweaked for best Q_{total} performance. NetTurnP BT426 7-fold is referring to a 7-fold cross-validation performed on the BT426 dataset. The other methods are as follows: DEBT (Kountouris and Hirst (2010)), E-SSpred (Liu et al. (2009)), BTNpred (Zheng and Kurgan (2008)), SVM (Hu and Li (2008)), MOLEBRNN (Kirschner and Frishman (2008)), BTSVM (Pham et al. (2003)), BetaTPred2 (Kaur and Raghava (2003)), COUDES (Fuchs and Alix (2005)), KNN (Kim (2004)), BTPRED (Shepherd et al. (1999)) and 1-4 and 2-3 correlation model (Zhang and Chou (1997)).
doi:10.1371/journal.pone.0015079.t001

β-turn-S networks

Classification networks were trained to predict whether an amino acid was located in one of the nine types of β-turn, as earlier defined. Networks were trained using the same method as described for β-turn-G i.e. an ensemble of 100 networks architectures for the first layer and 10 architectures for the second layer networks. Evaluation performances for the second layer β-turn-S networks are summarized in Table 3.3, along with a comparison against four other methods.

Evaluation of NetTurnP method against PLP datasets

Sequences for each of the three datasets PLP399, PLP364 and PLP273 were submitted to the NetTurnP, NetTurnP-tweak and the BetaTPred2 web-servers. Evaluation performances are summarized in Table 3.4.

Evaluating NetTurnP and NetTurnP-tweak showed that both methods are very stable over all three datasets, with only 0.22% difference in Q_{total} for NetTurnP, and 0.15% for NetTurnP-tweak within the datasets. The same

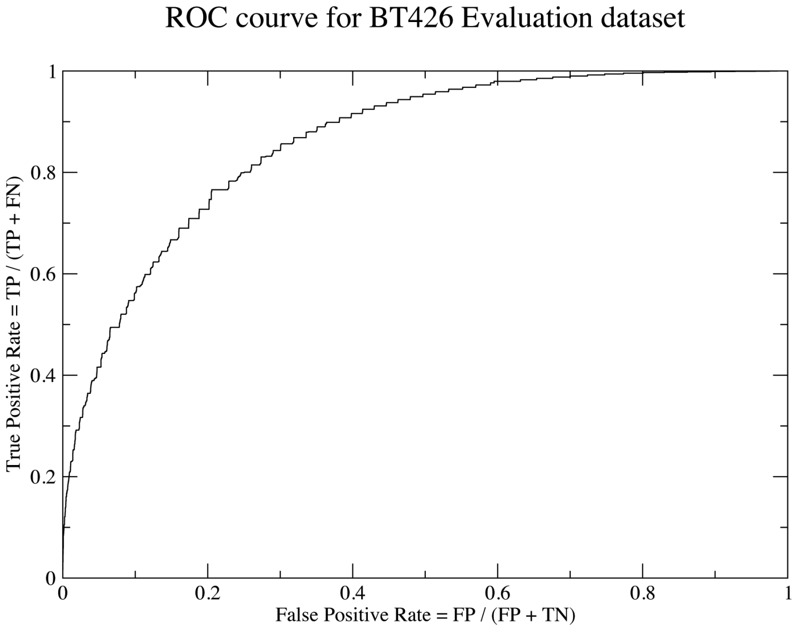


Figure 3.12. ROC curve for the evaluation of NetTurnP. The figure shows the ROC curve (True positive rate vs. False Positive Rate) for the evaluation of the NetTurnP against the BT426 dataset. doi:10.1371/journal.pone.0015079.g003

trend of stable prediction is seen for all other performance measures as well. NetTurnP and NetTurnP-tweak have a small decrease of 0.03 in MCC compared to the performance against BT426 (Table 3.2) whereas BetaTPred2 has an even bigger decrease of 0.06 in MCC. This indicates that both of the NetTurnP methods are still better than the BetaTPred2 method and now by an even bigger margin. Also, the slightly reduced MCC values indicate that the new PLP datasets contain more difficult targets compared to the original BT426 dataset.

Discussion

In the work presented in this paper a neural-network method called NetTurnP was developed. It predicts β-turns in general and the specific type of β-turn. This work represents one of the few studies where an independent evaluation dataset was used in addition to cross-validation. The evaluation set was non-homologous to the training datasets used. NetTurnP reached a Q_{total} of 78.2% with a MCC of 0.50, using a two-layered network structure, where the predictions from the first layer networks were used as input for the second layer.

Table 3.3. Comparison of NetTurnP and other β-turn methods for prediction of specific β-turn types

β-turn type	**Method**				
	MOLEBRNN	**COUDES**	**BETATURNS**	**DEBT**	**NetTurnP**
Type I	0.317	0.309	0.29	**0.36**	**0.36**
Type I'	**0.356**	0.226	N/A	N/A	0.23
Type II	**0.339**	0.302	0.29	0.29	0.31
Type II'	0.137	0.106	N/A	N/A	**0.16**
Type IV	0.236	0.109	0.23	**0.27**	**0.27**
Type VIII	0.109	0.071	0.02	0.14	**0.16**

The table shows a comparison of NetTurnP with other methods for prediction of β-turn types using the BT426 dataset. Performance values are given as Matthews correlation coefficients and the best are highlighted in bold. The methods are: MOLEBRNN (Kirschner and Frishman (2008)), COUDES (Fuchs and Alix (2005)), BETATURNS (Kaur and Raghava (2004)) and DEBT (Kountouris and Hirst (2010)) have all used seven-fold cross validation. We choose to completely exclude those data from the NetTurnP test and training and thus report evaluation performances against the BT426 dataset. The β-turn types VIII, V1a1 and VIa2 can only be predicted with correlations coefficients below or close to 0.1.
doi:10.1371/journal.pone.0015079.t002

Table 3.4. Evaluation of β-turn prediction on new PLP datasets

Prediction method	**Dataset**	**Q_{total}**	**PPV**	**Sens**	**Spec**	**MCC**	**AUC**
NetTurnP	PLP399	78.73	52.16	69.82	81.33	0.47	0.845
	PLP364	78.83	52.07	70.23	81.32	0.47	0.847
	PLP273	78.95	51.91	70.03	81.49	0.47	0.846
NetTurnP-tweak	PLP399	82.59	67.10	44.86	93.59	0.45	0.845
	PLP364	82.66	66.67	45.32	93.45	0.45	0.847
	PLP273	82.74	66.40	45.04	93.50	0.45	0.846
Betatpred2	PLP399	74.90	45.91	62.98	78.37	0.37	N/A
	PLP364	75.01	45.84	63.20	78.42	0.38	N/A
	PLP273	75.17	45.67	62.52	78.78	0.37	N/A

The table shows a comparison of NetTurnP, NetTurnP-tweak and the Betatpred2 method (Kaur and Raghava (2003)). The datasets PLP364 and PLP273 are subsets of PLP399, where PLP364 contain sequences deposited in PDB from 2008-2010 and PLP273 only contain sequences deposited from 2009-2010.
doi:10.1371/journal.pone.0015079.t003

β-turns tend to be located at solvent-exposed surfaces. Analyzing our training dataset (Cull-2220), we found that the most frequently observed amino acids in β-turns compared to the amino acid at any position were: Gly (11.6%/7.2%), Asp (8.9%/5.9%), Ser (7.1%/6.1%), Pro (7.0%/4.6%), Ala (6.4%/7.8%), Asn (6.3%/4.2%) and Glu (6.3%/7.0%). These amino acid residues are hydrophilic or small, where Pro is special due to its fixed and rigid structure making it suitable to reverse the direction of a protein chain. It is seen that Gly, Asp, Ser, Pro and Asn are occurring more often in β-turns than in general, and that Ala and Glu occur less frequently. A complete table of the frequencies for all amino acids is shown in Supplementary Table S6.

For the second layer networks different setups were tested in order to find the highest test (MCC) performance. We found that it was most optimal to use predictions from the networks β-turn-G and β-turn-P and with inclusion of predicted secondary structure and relative surface accessibility predictions.

The second layer networks were found to filter out the noise and increase the AUC value from 0.846 to 0.849 in test performance. This increase was found to be a significant increase corresponding to a p-value $\ll 0.001$, using an unpaired test with two independent samples (Armitage et al. (2002)). For the evaluation dataset BT426 the AUC increased from 0.860 to 0.864 after primary and second level networks, respectively. (p-value $\ll 0.001$).

Because of the unbalanced dataset (25% β-turns), Q_{total} is a poor measure by itself, as it is possible to achieve a Q_{total} of 75% if all residues were predicted to be non-β-turns. Instead, NetTurnP was trained to achieve the best MCC, which will also balance the performance measured on sensitivity and specificity. The effect of a tweaked Q_{total} performance (NetTurnP-tweak) showed that we could obtain a better Q_{total} than any other method, but at the expense that more false and true positives are removed as seen in Table 3.2 and Figure 3.13. Therefore only the most confident predictions remain, but the method becomes less sensitive. NetTurnP, with tweaking Q_{total}, achieves the best MCC performance of 0.48 compared to other methods.

For the prediction of specific β-turn types NetTurnP showed improved performance for four out of six β-turn types compared to other methods as seen in Table 3.3. We do provide a prediction via the webserver for the β-turn types VIb, VIa1 and Via2, even though the performances are quite low with MCC values of 0.11, 0.07 and 0.03 respectively. It is most likely due to the scarce number of these β-turn types.

Three new datasets were created with the purpose of evaluating NetTurnP and NetTurnP-tweak against a more recent set of sequences than the original dataset BT426. For the comparison NetTurnP/NetTurnP-tweak, DEBT, MOLEBRNN and BetaTPred2 were chosen. Due to errors in the DEBT and MOLEBRNN webservers, we were not able to obtain enough results for a comparison. MOLEBRNN never completed any calculations, and DEBT only succeeded to return a few results. All sequences were successfully submitted to NetTurnP/NetTurnP-tweak and BetaTpred2. Multiple sequences can be submitted to the NetTurnP webserver, which is a functionality that none of the other webservers provide.

For NetTurnP/NetTurnP-tweak the performance drops by 0.03 in terms of MCC compared to the performance obtained using the BT426 dataset. BetaTPred2 had an even bigger decrease in MCC of 0.06. This could indicate that the newer sequence data is a more challenging dataset.

A dataset of 75 experimentally determined antigen-antibody structures with predicted epitope residues was downloaded from the Supplementary

Section of DiscoTope (Haste Andersen et al. (2006)) in order to analyze the frequency of β-turns in discontinuous B-cell epitopes. We find that there is an overrepresentation with a factor 2 (data not shown) of β-turns in the discontinuous B-cell epitopes. We therefore believe that prediction of β-turns in general, can further improve immunological feature predictions.

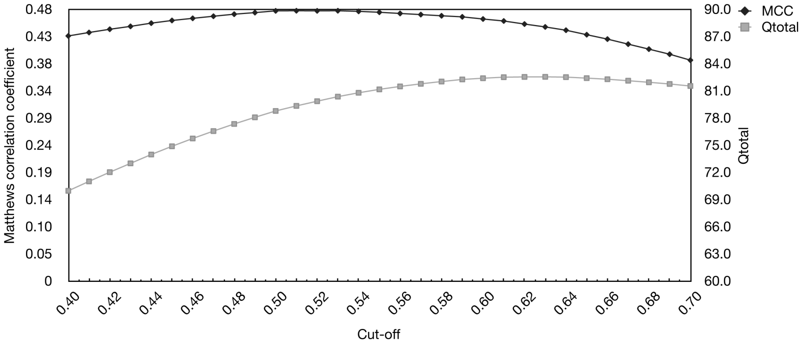


Figure 3.13. MCC and Q_{total} as function of the cut-off value. The figure shows MCC and Q_{total} as function of the cut-off value. The values are obtained by cross-validation of the Cull-2220 dataset. The X-axis is the threshold for a positive prediction of a β-turn. Y-axis to the left is the Matthews correlation coefficient and to the right Q_{total} values. doi:10.1371/journal.pone.0015079.g004

Materials and Methods

Evaluation dataset, BT426

To evaluate the NetTurnP method, a dataset of 426 non-homologous protein chains was used. The dataset, commonly known as BT426, was created by Guruprasad and Rajkumar (Guruprasad and Rajkumar (2000)) and consists of >94,800 amino acids. Several groups use it as a golden set of sequences upon which performance values are reported and compared. The dataset consists of protein chains whose structure has been determined by X-ray crystallography at a resolution of 2.0 Å or better. Each chain contains at least one β-turn region. In total 23,580 amino acids, corresponding to 24.9% of all amino acids, have been assigned to be located in β-turns. None of the sequences in the dataset share more than 25% sequence identity. The BT426 dataset was downloaded from the Raghava Group's website: http://www.imtech.res.in/raghava/bteval/dataset.html. Four sequences are obsolete from PDB and superceded by newer sequence data. Therefore 1GDO.A was replaced by 1XFF.A, 5ICB by 1IG5, 1ALO by 1VLB and 3B5C by 1CYO. This dataset was solely used for the final evaluation of our NetTurnP method.

Evaluation datasets, PLP399, PLP364 and PLP273

Three new datasets were constructed with the purpose of evaluating the NetTurnP method against a more recent set of protein sequences. Protein sequence data was extracted from the RCSB (Research Collaboratory for Structural Bioinformatics) Protein Data Bank (PDB) (Berman et al. (2000)) using the protein culling server PISCES (Wang and Dunbrack Jr (2003)). An initial dataset was created using the following criteria: Maximum sequence identify <= 25%, Resolution <= 2.0 Å , R-factor <= 0.2, sequence in the range 25 - 10,000 amino acids and including X-structures only. The resulting dataset contained 3,572 PDB protein chains before homology reduction. A Hobohm1 algorithm with a threshold as described previously (Lund et al. (1997)) was used to create the final homology reduced dataset PLP399, containing 399 protein chains. No sequences in the dataset share more than 25% sequence identity to a sequence within the BT426 dataset, Cull-2220 dataset (described below) or the datasets used for training and evaluation of the NetSurfP method (Petersen et al. (2009)). The PLP399 dataset was further subdivided into PLP364 containing only sequences with deposition date from 2008 and newer and PLP273 containing sequences from 2009-2010. All three datasets are solely used to evaluate the NetTurnP method, and they are available for download at http://www.cbs.dtu.dk/services/NetTurnP/suppl/plp.php.

Training dataset, Cull-2220

Protein sequence data was obtained from PDB using PISCES. A dataset was constructed in two steps, first an initial selection of potential sequences and later a more strict selection based on a Hobohm1 (Hobohm et al. (1992)) homology reduction algorithm. First PDB was culled using the following criteria: Maximum sequence percentage identity <= 40%, Resolution <= 3.0 Å R-factor <= 0.2, sequence length in the range 40 - 10,000 amino acids and including X-ray structures only. The resulting dataset contained 5,648 PDB protein chains before homology reduction. An empiric sorting function (Equation 3.1) was applied to rank the protein chains such that high-resolution structures with the most experimentally determined amino acids were preferred instead of the shorter low-resolution homologous protein sequence. A Hobohm1 algorithm with a threshold as described previously (Lund et al. (1997)) was used to create the final homology reduced dataset (Cull-2220). No sequences in the dataset share more than 25% sequence identity to a sequence within the BT426 dataset.

$$score = \frac{resolution^2 \times sequence\ length}{pdb\ length} \tag{3.1}$$

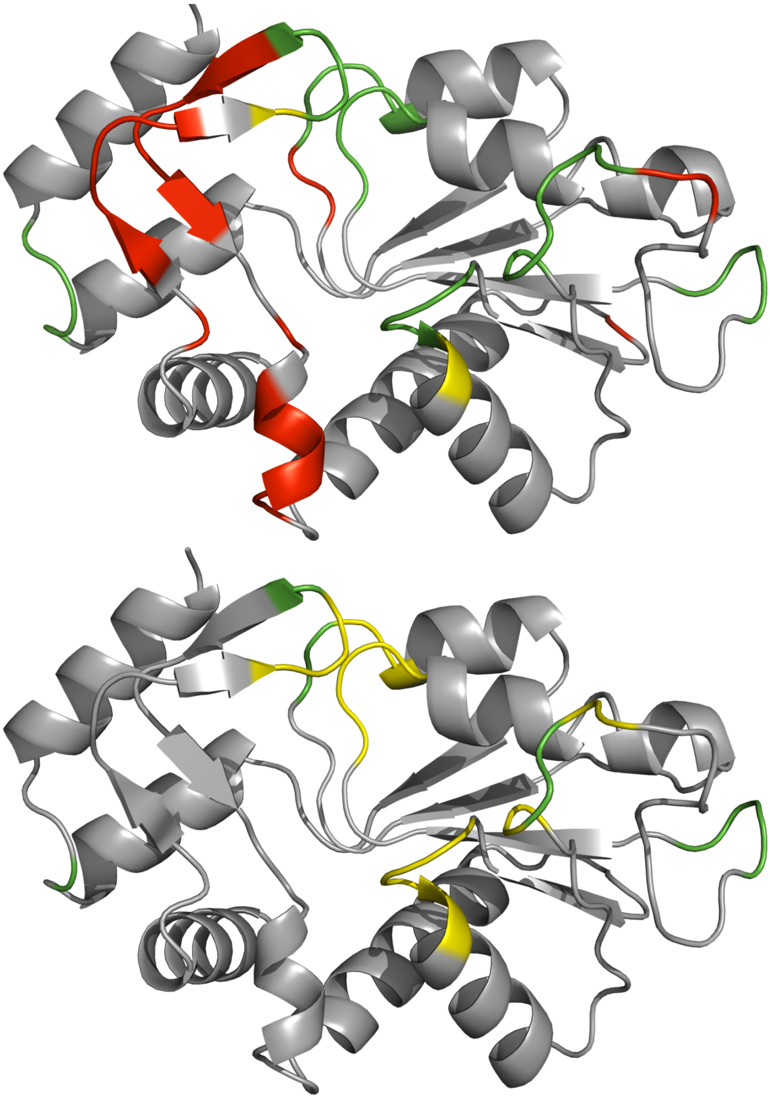


Figure 3.14. 1D projection of β-turn predictions for default and Q_{total} optimized cut-off plotted on 3D structure 2WNS chain A. The figure shows the structure of a transferase, 2WNS chain A. The top structure shows a prediction where default cut-off has been used (NetTurnP) and the bottom structure shows the same structure where cut-off tweak has been applied (NetTurnP-tweak). Assigned β-turns are yellow, false positives are red, and the residues in green are where assignments and predictions agree. Figures were made using the PYMOL software (DeLano (2002)). doi:10.1371/journal.pone.0015079.g005

Equation 3.1 – Ranking of experimentally determined protein sequences. The best rank is assigned to the protein sequence with the lowest score. ‘‘resolution’’ is the resolution in Ångstroms according to PDB, ‘‘sequence length’’ is referring to the actual length of the sequence which may include amino acids for which there are no available coordinates in the PDB-file. ‘‘pdb length’’ is the length of the sequence for which there are coordinates for the amino acids.

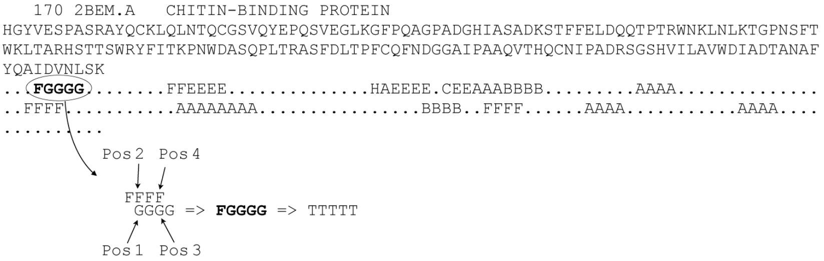

170 2BEM.A CHITIN-BINDING PROTEIN
HGYVESPASRAYQCKLQLNTQCGSVQYEPQSVEGLKGFPQAGPADGHIASADKSTFFELDQQTPTRWNKLNLKTGPNSFT
WKLTARHSTTSWRYFITKPNWDASQPLTRASFDLTPFCQFNDGGAIPAAQVTHQCNIPADRSGSHVILAVWDIADTANAF
YQAIDVNLSK
..FGGGG........FFEEEE...............HAEEEE.CEEAAABBBB.........AAAA..............
..FFFF...........AAAAAAAA................BBBB..FFFF......AAAA...........AAAA....
....
Pos 2
Pos 4
FFFF
GGGG => FGGGG => TTTTT
Pos 1
Pos 3

Figure 3.15. Assignment scheme used to train the β-turn-P method. Figure 3.15 is illustrating the assignment scheme used to train the β-turn-P method for an example protein sequence with PDB-identifier 2BEM.A. A β-turn with a length of five shown as T’s, is composed of two overlapping β-turn types, here indicated with F (Type VIII) and G (Type VIa2). In this situation, one β-turn residue can be assigned as being both at position 1 and at position 2. Another β-turn residue can be assigned as being both at position 3 and at position 4. doi:10.1371/journal.pone.0015079.g006

β-turn assignment

The program PROMOTIF (Hutchinson and Thornton (1996)) was used for assignment of β-turns and for the Cull-2200 dataset where 98,624 out of 451,812 amino acids were assigned to be inside a β-turn (21.8%) region.

According to restraints on phi (Φ) and psi (Ψ) dihedral angles between residues *i+1* and *i+2*, nine β-turn type specific datasets were created. The Φ, Ψ restrains for each of the types (I, I’, II, II”, IV, VIII, VIb, VIa1 and VIab) are shown in Supplementary Table S7. These angles are allowed to deviate ± 30° from the defined angles, with the addition that one dihedral angle is allowed to deviate as much as ± 40°. Type VIa1 and VIa2 also require a cis-proline at position *i+2*.

For the general prediction of β-turns, the positive set includes the amino acid residues that belong to any of the 9 β-turn types and the negative set include all other residues. For the type specific β-turn predictions, the positive sets were reduced to include only β-turns of one specific type whereas everything else comprised a negative dataset. The number and percentage of amino acids (positive sets) in each of the type specific datasets are: type I 40,482/9.0%, type I' 4,812/1.1%, type II 14,375/3.2%, type II' 3,124/0.7%, type IV 38,445/8.5%, type VIII 11,192/2.5%, type VIb 1,120/0.3%, type VIa1 736/0.2% and type VIa2 214/0.1%.

Position Specific Scoring Matrices

Sequence profiles i.e. Position-Specific Scoring Matrices (PSSM) were generated for all protein chains, using the iterative PsiBLAST program (Altschul et al. (1997)). Query sequences were blasted for four iterations against a local copy of the National Center for Biotechnology Information (NCBI) non-redundant (nr) sequence database, which for speed purposes had been homology-reduced using CDHIT (Li et al. (2001)) to less than 70% sequence identity. An E-value cut-off of 1×10^{-5} was used.

Secondary structure and surface accessibility

Secondary structure and surface accessibility predictions were generated for all protein chains, using the NetSurfP program (Petersen et al. (2009)).

Neural Networks

A standard feed-forward procedure was utilized to train the neural networks (Rumelhart et al. (1986)), and a gradient descent method was used to back-propagate the errors where-after weights were updated (Lund et al. (2005)). A sliding window of amino acids was presented to the neural network and predictions were made for the central position. The neural networks were trained using window sizes of 5, 7, 9, 11 and 13, the following number of hidden units: 50, 75, 100 and 125, and two output neurons. Altogether we used 20 different neural network architectures. A 10-fold cross-validation procedure was used, thus a total of 200 neural networks. Synapse weights were stored for the epoch where the best test set Matthews correlation coefficient was obtained.

Amino acids were encoded both using PSSM values, three neurons for predicted helix, strand and coil and one extra neuron for the relative surface accessibility, thus a total of 25 neurons were used to describe an amino acid.

Optimized Networks

Three different types of artificial neural networks have been trained:

- β-turn-G
- β-turn-S
- β-turn-P

The β-turn-G (G for general) method predicts if an amino acid is located in a β-turn region or not.

The β-turn-S (S for specific) method was trained to predict if an amino acid belongs to any of the nine β-turn classes: I, I', II, II', IV, VIII, VIb, VIa1 and VIab.

The method β-turn-P (P for position) is a combination of four sub-methods that have been trained to predict if an amino acid is located at position 1, position 2, position 3 or position 4 in a β-turn. Some amino acids can be assigned to multiple positions within a β-turn as shown in Figure 3.15. However, within each of the four sub-methods only one position was considered.

We found that the performance of the methods β-turn-G and β-turn-S could be improved by use of a second layer of neural networks where information from the β-turn-P method was included as input. A second layer is often used as some of false predictions can be corrected (Petersen et al. (2009), Petersen et al. (2000)) and is due to the fact that new or enriched input data is provided for the second layer neural networks.

Performance measures

The quality of the predictions was evaluated using six measures; Matthews correlation coefficient (Matthews (1975)) (MCC), Q_{total}, Predicted Positive Value (PPV), sensitivity, specificity and Area under the Receiver Operating Curve (Swets (1996)) (AUC). FP = False Positive, FN = False Negative, TP = True Positive, TN = True Negative.

$$MCC = \frac{(TP \times TN - FP \times FN)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}} \tag{3.2}$$

Matthews correlation coefficient can be in the range of -1 to 1, where 1 is a perfect correlation and -1 is the perfect anti-correlation. A value of 0 indicates no correlation.

$$Q_{total} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.3}$$

Q_{total} is the percentage of correctly classified residues, also called the prediction accuracy.

$$PPV = \frac{TP}{TP + FP} \times 100 \tag{3.4}$$

PPV is the Predicted Positive Value, also called the precision or Q_{pred}.

$$Sensitivity = \frac{TP}{TP + FN} \times 100 \tag{3.5}$$

Sensitivity is also called recall or Q_{obs}, and is the fraction of the total positive examples that are correctly predicted.

$$Specificity = \frac{TP}{TP + FP} \times 100 \tag{3.6}$$

Specificity is the fraction of total negative examples that are correctly predicted.

The above-mentioned performance measures are all threshold dependent and in this work a threshold of 0.5 was used, unless otherwise stated.

AUC is a threshold independent measure, and was calculated from the ROC curve which is a plot of the sensitivity against the False Positive rate = FP/(FP + TN). An AUC value above 0.7 is an indication of a useful prediction and a good prediction method achieves a value >0.85 (Lund et al. (2005)).

Supporting Information

Supplementary Table S3 Setups tested for training in the second layer networks. The table is listing the different setups tested for training in the second layer networks. In the table abbreviations are as follows: β-turn-G = β-turn/not-β-turn prediction from first layer networks, β-turn-P = position specific predictions from first layer networks, sec-rsa = secondary structure and surface accessibility predictions from NetSurfP (Petersen et al. (2009)), PSSM = Position Specific Scoring Matrices.
(DOCX)

Supplementary Table S4 Test performance for the first layer β-turn-P networks. Test performances from the first layer β-turn-P networks using the Cull-2220 dataset. All performance measures have been explained in the methods section. All β-turn-P networks were trained using pssm + sec + rsa, where pssm = Position Specific Scoring Matrix, sec = Secondary structure predictions (Petersen et al. (2009)), rsa = Relative solvent accessibility predictions (Petersen et al. (2009)). The positions in the four network trainings are referring to the position in a β-turn.
(DOCX)

Supplementary Table S5 Test performances from the first and second layer β-turn-G networks using the Cull-2220 dataset. All performance measures have been explained in the methods section. The first layer networks were using pssm + sec + rsa, and the secondary networks were using β-turn-P + β-turn-G + sec + rsa, where the used nomenclature are: pssm = Position Specific Scoring Matrix, sec = secondary structure predictions (Petersen et al. (2009)), rsa = relative solvent accessibility predictions (Petersen et al. (2009)). β-turn-G = β-turn/non-β-turn predictions, β-turn-P = predictions from the position specific networks.
(DOCX)

Supplementary Table S6 Amino acid statistics in Cull-2200 dataset. Frequencies for amino acids in β-turns and the Cull-2220 training set. The first part of the table 'β-turn statistics' shows the amount of residues, which have been assigned as β-turns and their percentage of the total amount of β-turn assigned residues in the Cull-2220 set. The second part of the table 'Amino acid statistics' shows the amount of residues and the percentage of the total Cull-2220 set.
(DOCX)

Supplementary Table S7 Dihedral angles for the β-turn types as used by PROMOTIF. Dihedral angles for the β-turn types between residues two (*i+1*) and three (*i+2*) as used by PROMOTIF (Hutchinson and Thornton (1996)). These angles are allowed to deviate by ± 30° from the defined angles, with the addition that one dihedral angle is allowed to deviate as much as ± 40°. Type IV is used for all β-turns, which do not fall within the dihedral angle ranges for the eight defined types. Type VIa1, VIa2 also require a cis-proline at position *i+2*.
(DOCX)

Supplementary Figure S1 Matthews correlation using different setups and an increasing number of trained network architectures. The figure shows test performances in Matthewss correlation coefficient when including an increasing number of trained networks architectures, named *Top ranked network architectures*, based on test set performance using different setups. Abbreviations for the setups are as follows: β-turn-P = position specific first layer predictions, β-turn-G = general β-turn/not-β-turn first layer predictions, sec-rsa = secondary structure and surface accessibility predictions from NetSurfP (Petersen et al. (2009)), PSSM = Position Specific Scoring Matrices. The setups are composed as follows: A = PSSM + sec-rsa, B = PSSM + β-turn-G + sec-rsa, C = PSSM + β-turn-G, D = PSSM + β-turn-P, E = β-turn-P, F = β-turn-G + sec-rsa, G = β-turn-G, H = PSSM + β-turn-P + sec-rsa, I = β-turn-P + sec-rsa, J = PSSM + β-turn-P + β-turn-G + sec-rsa, K = PSSM + β-turn-P + β-turn-G, L = β-turn-P + β-turn-G, M = β-turn-P + β-turn-G + sec-rsa.
(TIFF)

Acknowledgments

We would like to thank Ramneek Gupta with his help to proofread this manuscript.

Author Contributions

Conceived and designed the experiments: BP CL TNP. Performed the experiments: BP CL TNP. Analyzed the data: BP CL TNP. Contributed reagents/materials/analysis tools: BP CL TNP. Wrote the paper: BP CL TNP.

↺ ↻ ↺

Chapter 4

Caching of data

The caching project was initialized due to the reason that NetSurfP had been included in the EPipe project (http://www.cbs.dtu.dk/services/EPipe/, Blicher et al. (2010)). EPipe is a project with an automated web-server performing comparative analysis of protein sequences. It offers several different features and prediction tools to be included in the analysis, and returns a detailed report with the complete analysis and prediction results. After the inclusion of NetSurfP in the pipeline, NetSurfP had become a bottleneck for the web-server due to the long calculation time caused mainly by the calculation of the PSSM. It was therefore decided to pre-calculate and cache both the PSSM and the final NetSurfP predictions. Caching the data would not only speed up NetSurfP, but also all other servers using PSI-BLAST generated sequence profiles or NetSurfP predictions.

Several data formats/outputs are stored in the database as can be seen in Table 4.1. Due to the fact that many human sequences are submitted to the EPipe server (personal communication), the proteome for *Homo Sapiens* build 25 was downloaded from EBI (http://www.ebi.ac.uk) and precalculated both as sequence profiles, NetSurfP and NetTurnP formats.

As a benchmark of NetSurfP with and without caching enabled, 1,000 randomly chosen human protein sequences from the 25.H release were selected. The smallest sequence contained 15 amino acids and the largest sequence contained 33,615 amino acids. All sequences were submitted to the in-house version of NetSurfP and the total compute time was calculated when caching was disabled and when the cached sequence profiles or final NetSurfP output was used. As it can be seen from Table 4.2 there is a huge speed increase when caching is enabled. It took a total of 77 hours and 20 minutes for NetSurfP to complete all 1,000 sequences with caching disabled, where it took 9 hours and 42 minutes when the caching was enabled. Note, in both cases only one

Table 4.1. Data formats/outputs and amount of records stored in the cache

Data	Records stored
PSSM	543,013
netsurfp-1.1	526,422
netturnp-1.0|TNT	346,711
netturnp-1.0|typeA	341,695
netturnp-1.0|typeB	341,291
netturnp-1.0|typeC	341,259
netturnp-1.0|typeD	341,235
netturnp-1.0|typeE	341,237
netturnp-1.0|typeF	341,201
netturnp-1.0|typeG	341,187
netturnp-1.0|typeH	341,167
netturnp-1.0|typeI	341,185

The table lists the data formats/outputs and number of records saved in the cache as of March 31, 2011.

Table 4.2. Benchmark of NetSurfP with and without caching enabled, using 1,000 randomly chosen human proteins

Method	Time
NetSurfP 1,000 sequences, no caching	77 hours and 20 minutes
NetSurfP 1,000 sequences, PSSM cached	9 hours and 42 minutes
NetSurfP 1,000 sequences, NetSurfP cached	0 hours and 14 minutes

1,000 human proteins were randomly selected from the human proteome build 25.H and submitted to NetSurfP with caching either enabled or disabled with no parallel computation. The first row shows the total compute time when caching is disabled. The second row shows the compute time where PSSM's were cached and NetSurfP output was recomputed for each sequence. The third row indicates how fast the computation approximately will be when NetSurfP output is fetched from the cache.

sequence was processed at a time. Due to the way the NetSurfP source code is written, there is still room for speed improvements. When the caching system was implemented in the NetSurfP source code, it was only installed for the generation of the sequence profiles, and there is no direct look-up for the NetSurfP output. The reason for the long calculation time of NetSurfP is that the sequence is processed over all the synapse files, before a result is returned. As can be seen from the last row in Table 4.2, it only took $\sim$ 14 minutes to get a result when the raw NetSurfP output was fetched and processed. It shows that there are room for improvements in terms of speed and this will be implemented at a later stage. All calculations were performed on a titanium ia64 GNU/Linux machine.

Chapter 5

HIV-1 protease substrate specificity

The following chapter describes the work performed at the University of Copenhagen in the lab of Martin Willemoës at the Department of Biology in 2010. An attempt to predict the specificity of the HIV-1 Protease will be presented.

The motivation for starting this project stems from a focus area in the group of Martin Willemoës, where they want to modify a HIV-1 protease in order to cleave other desired sequences, hence becoming a valuable tool for biological applications. This requires the rational redesign of the enzyme, to achieve new enzymatic activity towards a specific sequence and such a process should be driven by an interplay between experimental and computational work. If successful, the rational redesign could be extended to any sequence. My part of the project was to develop a HIV-1 protease specificity predictor, to identify variants of known substrates and to identify new substrates. A selection of newly identified substrates will then be synthesized and verified in the lab of Martin Willemoës.

5.1 Background

5.1.1 HIV-1 protease

More than 33.3 million people are infected with HIV (2009 estimates), whereas a total of 2.6 million became infected in 2009 alone. More than 1.8 million people died in 2009 due to Acquired Immuno Deficiency Syndrome - AIDS (http://www.unaids.org/). HIV is the virus causing AIDS and it exists in two variants, type 1 (HIV-1) and type 2 (HIV-2), where HIV-1 is the most prevalent type in the worldwide pandemic. HIV-1 can further be classified into groups, subtypes and sub-subtypes. One of the common characteristic for all of them is that they contain the HIV-1 protease (Clemente et al. (2006)). The HIV-1 protease (PR) is a retroviral aspartyl protease, which is completely essential for the life-cycle of HIV. A protease is an enzyme conducting proteolysis, which is the degradation of a protein. PR processes the Gag and Gag-Pol polyproteins into structural and functional proteins. The *pol* gene is responsible for the basic mechanisms for which the virus can reproduce, for example, the HIV protease, integrase and reverse transcriptase. The *gag* gene is responsible for the basic physical infrastructure of the virus, of which matrix protein, capsid protein and nucleocapsid protein can be mentioned.

The understanding of the PR is important in the fight against HIV, due to the reason that a non-functional PR results in immature and non-infectious HIV virus particles (Nalam et al. (2010)). PR is therefore an important target for the rational design of drugs against HIV. A handful of commercially available drugs already exists, for example Saquinavir, Ritonavir, Indinavir and Nelfinavir (Wensing et al. (2010)). They are all PR inhibitors, which inhibit the activity of the protease.

The PR is a protein consisting of two symmetrical subunits, each of them having the length of 99 amino acids. These two subunits form a tunnel with two flaps in the top of the structure as can be seen in Figure 5.1. The flaps can move and thereby allow or disallow proteins to enter the tunnel. When a protein has entered the tunnel, the flaps move in order to fine tune the interaction between the PR and the bound substrate where after it is cleaved.

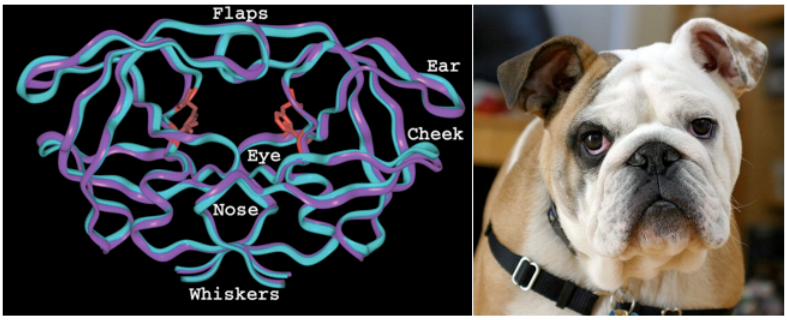


Figure 5.1. The figure shows the two symmetrical subunits of the HIV-1 protease labelled according to its resemblance to an English bulldog. Cyan ribbon is the backbone of a drug resistant mutant with PDB identifier 1D4S and the purple ribbon displays a wild-type with PDB identifier 1KZK. Picture courtesy of Perryman et al. (2004)(left picture) and Michael Lazarev[2] (right picture).

[2] http://www.ehow.com/facts_5196353_english-bulldog.html 18 March 2011

The HIV-1 protease recognizes peptides of 8 amino acids in length, which will from now on be named substrates. They are usually represented as $P_4 - P_3 - P_2 - P_1 \downarrow P_{1'} - P_{2'} - P_{3'} - P_{4'}$ (Schechter and Berger (1967)), where ↓ denotes a scissile bond, the cleavage site. P_n denotes an amino acid from the N-terminus and $P_{n'}$ from the C-terminus of the substrate. The 12 substrates recognized by PR are listed in Table 5.1, and the resulting proteins are shown schematically in Figure 5.2, where the cleavage sites are listed at the Gag and Gag-Pol polyproteins.

Table 5.1. Recognition sequences cleaved by HIV-1 protease

Substrate sequence	**Cleavage domain**
Cleavage sites in *gag*:	
SQNY↓PIVQ	MA-CA
ARVL↓AEAM	CA-p2
ATIM↓MQRG	p2-NC
RQAN↓FLGK	NC-p1
RQAN↓FLRE	NC-TFP
PGNF↓LQSR	p1-p6gag
Cleavage sites in *pol*:	
DLAF↓LQGK	TFP-p6pol
SFNF↓PQVT	p6pol-PR
TLNF↓PISP	PR-RTp51
AETF↓YVDG	RTp51-RTp66
RKVL↓FLDG	RTp66-INT
DCAW↓LEAQ	NEF

The table lists the substrate sequences recognized by HIV-1 protease and their cleavage domains (Perez et al. (2010)). The ↓ defines the cleavage site.

5.1.2 HIV-1 protease specificity prediction

The key towards developing inhibitors of this protease will come from a better understanding for the specificity of the HIV-1 protease. Several methods have been applied in the approach for predicting the specificity of the PR, of which the following can be mentioned. Artificial Neural Networks, achieving a classification accuracy of 88% on a test set of 39 samples (Thompson et al. (1995)), three years later Cai et al. (1998) repeated the before mentioned work and reached an accuracy of 92% on a test set of 63 samples. Support Vector Machines (SVM) have also been applied with a prediction accuracy of 87% on the same test set of 63 samples (Cai et al. (2002)) and further explorations on different data sets showed that prediction of PR specificity is a non-linear problem suitable for SVM (Rögnvaldsson and You (2004)).

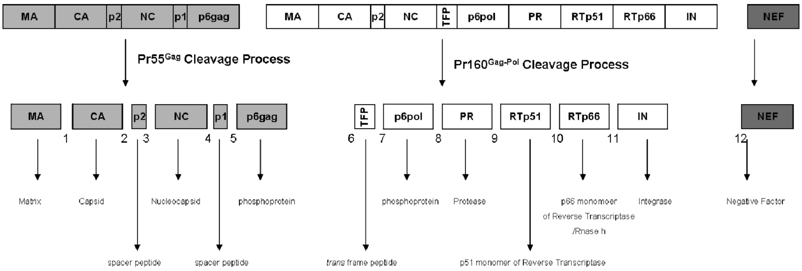


Figure 5.2. The figure shows a schematic representation of the Gag (light gray) and Gag-Pol (white) polyproteins with the 12 individual protease cleavage sites (Perez et al. (2010)).

Recently a web-server, HIVCleave, for the prediction of cleavage sites both in HIV-1 and HIV-2 has been developed (Shen and Chou (2008)).

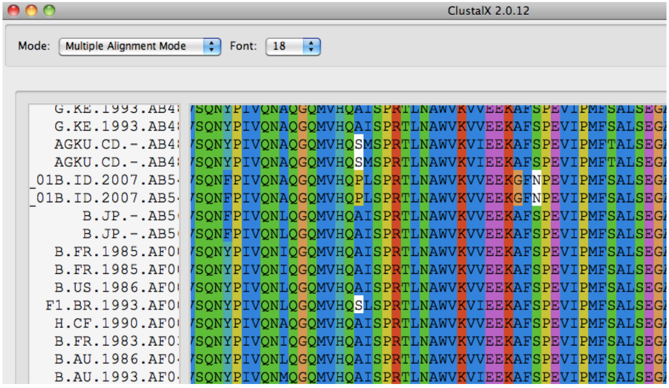
Although many studies have been carried out in this area, no perfect rule has yet been found to determine whether or not a peptide is cleaved by the HIV-1 Protease. Some PR inhibiting drugs do exist, but due to the high mutation rate of retroviruses, there is always a risk that the selective pressure will change amino acids in the PR. Thereby PR's specificity could be changed, which would make the drugs useless. For that reason drugs administered against HIV are often given in cocktails of three to four drugs, each if them targeting different stage of the HIV life cycle.

Because of the high mutation rate of retroviruses there is a need for new broad range PR inhibitors, which can compete with future mutants. In order to find new potential inhibitors, knowledge about the specificity of substrates is essential. An ideal protease inhibitor should have a well-defined substrate specificity, which is broad enough to treat the disease efficiently, but in the meantime so narrow that it does not interfere with the other proteases in the body. In the search for new substrates predictions are needed to narrow down the candidates, since the number of possible 8-mers is an astronomical number and would therefore be impossible to test in lab experiments. Using prediction methods a handful of the best candidates can then be selected and tested by carrying out experiments in the lab. Until now the HIV-1 Protease specificity is only partially understood, due to the reason that the cleavage sites do not share any obvious sequence homology or binding motifs.

5.2 Materials and Methods

5.2.1 Data preparation - positive dataset

The sequence for HIV-1 (HXB2) with accession number K03455 was downloaded from GenBank in DNA format and thereafter translated to amino acid sequences. HXB2 is a sequence derived from the first HIV-1 isolate (Hahn et al. (1984)) and is a standard reference strain. Protein sequences



Mode: Multiple Alignment Mode Font: 18

```
  G.KE.1993.AB4  SQNYPIVQNAQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  G.KE.1993.AB4  SQNYPIVQNAQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  AGKU.CD.-.AB4  SQNYPIVQNAQGQMVHQSMSPRTLNAWVKVIEEKAFSPEVIPMFTALSEG
  AGKU.CD.-.AB4  SQNYPIVQNAQGQMVHQSMSPRTLNAWVKVIEEKAFSPEVIPMFTALSEG
_01B.ID.2007.AB5 SQNFPIVQNAQGQMVHQPLSPRTLNAWVKVVEEKGFNPEVIPMFSALSEG
_01B.ID.2007.AB5 SQNFPIVQNAQGQMVHQPLSPRTLNAWVKVVEEKGFNPEVIPMFSALSEG
     B.JP.-.AB5  SQNFPIVQNLQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
     B.JP.-.AB5  SQNFPIVQNLQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.FR.1985.AF0  SQNYPIVQNIQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.FR.1985.AF0  SQNYPIVQNIQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.US.1986.AF0  SQNYPIVQNLQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
 F1.BR.1993.AF0  SQNYPIVQNLQGQMVHQSLSPRTLNAWVKVIEEKAFSPEVIPMFSALSEG
  H.CF.1990.AF0  SQNYPIVQNAQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.FR.1983.AF0  SQNYPIVQNIQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.AU.1986.AF0  SQNYPIVQNLQGQMVHQAISPRTLNAWVKVVEEKAFSPEVIPMFSALSEG
  B.AU.1993.AF0  SQNYPIVQNMQGQMVHQAISPRTLNAWVKVIEEKAFSPEVIPMFSALSEG
```

Figure 5.3. Visually inspection of substrate SQNY↓PIVQ in ClustalX.

for Gag, Pol and Nef were downloaded from the Los Alamos HIV sequence Database (http://www.hiv.lanl.gov/)(LAH), and blasted against HXB2 to make sure that they align perfectly with the reference genome. The HIV-1 Protease with PDB identifier 1HXB was downloaded from PDB and blasted against HXB2 to make sure it was the right protease. All 12 substrate sites were identified in the model sequences for Gag, Pol and Nef. All genomes matching virus type HIV-1 were downloaded from LAH, and filtered to make sure they contained the right protease 1HXB. Sequences were aligned separately against the specific substrate sites in Gag, Pol and Nef using the alignment program MAFFT (Katoh et al. (2002)). Sequences that had too many gaps, frameshifts, stopcodons or in general were impossible to align, were discarded. Three substrates, ATIM↓MQRG, RQAN↓FLRE and PGNF↓LQSR, were found impossible to align properly and were therefore not used for further analysis.

All alignments were visually inspected using the alignment program ClustalX (Thompson et al. (1997), Larkin et al. (2007)). An example of substrate SQNY↓PIVQ visualised in ClustalX is presented in Figure 5.3, where a mutation from 'Y' to 'F' can be seen in four of the sequences. Peptides with length of eight amino acids were cut out from the substrate alignments and used as the positive set of cleavable substrates. One positive set was created for each of the nine substrate sequences.

5.2.2 Negative dataset

Negative peptides were made by creating all possible 8-mers from the Gag and Pol polyproteins. All the peptides found in the positive sets, were excluded from the negative dataset. As a result 1,478 peptides were included in the final negative dataset.

5.2.3 Weight-Matrices

Weight matrices are calculated from an aligned set of sequences that shares a common motif, in this case it is substrates, which are being cleaved by the HIV-1 Protease. The weight matrix has a dimension of twenty by i, where i in this case is eight due to the eight amino acids in the substrate. Each cell contains the log-odds score for a given amino acid at position i as explained in Section 1.2. To correct for possible sampling biases, substrates are clustered using a 62% identity threshold as described by (Henikoff and Henikoff (1994)). It is done in order to down-weight the substrates, which show a high degree of similarity. A weight of 1/N is hereafter assigned to all substrates in the clusters, where N is the number of substrates in the cluster. Therefore none of the substrates is removed, but each of them has a lower weight in the statistics, when the matrix is generated. To make sure all amino acids are represented in the dataset pseudocount correction was used (Altschul et al. (1997)).

A low scoring peptide in the final weight matrix shows that this peptide is missing the feature, which was in the peptides that were used to generate the matrix, while a high scoring peptide indicates that this peptide is more likely to have this feature (Lund et al. (2005)).

Each of the previously mentioned nine positive substrate datasets was homology reduced so that only non-identical substrates were used in the creation of the weight matrices. All of the substrate datasets was assigned a test set number, as seen in Table 5.2, and weight matrices were thereafter created for all individual substrate datasets. In order to create average matrices (avemat), the average score for each position in the weight matrix was calculated using a 9-fold cross-validation scheme. For example, the avemat for test set 1 is an average of the matrices created for test sets 2 to 9, leaving out the matrix from test set 1. The reason for calculating average matrices is to ensure that all substrates are weighted equally. Performances are hereafter calculated using the peptides from the left-out test sets together with the peptides from the negative set. Sequence logos were generated for all substrate- and average matrices using an in-house sequence logo generator.

5.3 Results and Discussion

In this work, we present an attempt to predict the specificity of the HIV-1 Protease. Nine of the twelve substrates of which the HIV-1 Protease recognizes were selected as can be seen in Table 5.2. The table also shows that for most of the substrates, the given substrate sequence is also the highest ranking among the most frequently occurring variant. In two cases, DLAF↓LQGK and SFNF↓PQVT other variants are occurring more frequently than the given substrate sequence. Both of the substrates also have quite a high number of different variants, as seen in Table 5.2 column 2. For the DLAF↓LQGK substrate this motif is represented in 91 out of 1,972 sequences, where the most frequently occurring motif for that substrate, NLAF↓PQGE, is occurring 387

Table 5.2. Substrates with the amount of sequences, variants and their rank among the most frequently occurring variants.

Substrate sequence	# Variants	# Sequences	Rank
1: AETF↓YVDG	23	2,012	1
2: RQAN↓FLGK	46	5,208	1
3: DCAW↓LEAQ	266	5,736	1
4: SQNY↓PIVQ	54	5,282	1
5: RKVL↓FLDG	24	2,010	1
6: ARVL↓AEAM	33	5,392	1
7: DLAF↓LQGK	118	1,972	6
8: SFNF↓PQVT	98	2,003	21
9: TLNF↓PISP	20	2,009	1

The table lists the total amount of sequences for each of the examined substrates and the amount of variants found for each substrate. The last column show where the substrate ranks among the most frequently occurring variants.

times (data not shown). For the SFNF↓PQVT substrate the picture is quite different. Here this motif is only represented in 13 out of 2,003 sequences, while the most frequently occurring motif, SFSF↓PQIT, is represented in 628 sequences (data not shown).

Sequence logos were created for all substrate sequences, as seen in Figure 5.4. They are all very different concerning position conservation and information content. General for most of the substrates is, that there is a preference for hydrophobic residues. This is seen by the dominance of Ala, Phe, Leu, Ile, Pro and Val in the sequence logos. These are mostly residing at position 3, 4 and 5, and positions 2 and 7 can accommodate a variety of residues. This can also previously been found by Kontijevskis et al. (2007).

Substrate 2 (RQAN↓FLGK) and substrate 6 (ARVL↓AEAM) only have little information content available compared to the other substrates, which means that they have a higher variety of amino acids at the individual positions in the substrate sequences. For substrate 6 it seems that the positions with the most information are positions 3 and 4, which predominantly are hydrophobic residues. Substrate 3 (DCAW↓LEAQ) on the other hand have a high preference for Trp at position 4, and position 5 also seem to be more conserved with a preference for Leu. Substrate 4 (SQNY↓PIVQ), substrate 8 (SFNF↓PQVT) and substrate 9 (TLNF↓PISP) are all three dominated by Pro at position 5, and generally have a preference for hydrophobic amino acids. Substrate 1 (AETF↓YVDG) and substrate 5 (RKVL↓FLDG) both have a high preference for Asp at position 7 and Gly at position 8. Substrate 7 (DLAF↓LQGK) and substrate 9 (TLNF↓PISP) both share a preference for Leu and Phe at position 2 and 4.

Sequence logos were also generated for all 9-fold avemat, as seen in Figure 5.5. These logos are very similar and show a higher preference for Phe and

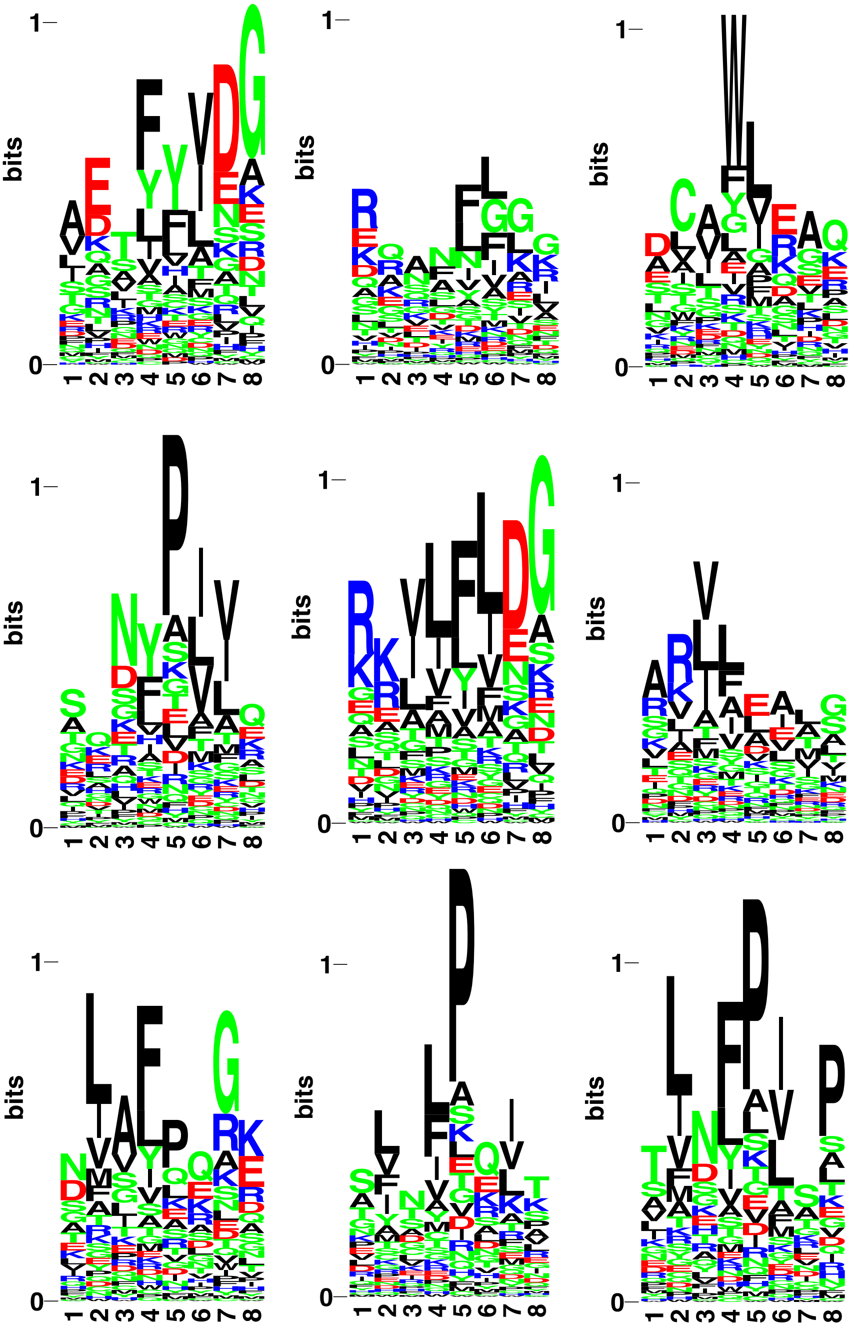


Figure 5.4. Sequence logos for each of the nine substrate sequences, where substrate 1 (referring to Table 5.2) is in the top left corner, and substrate 9 in the bottom right corner.

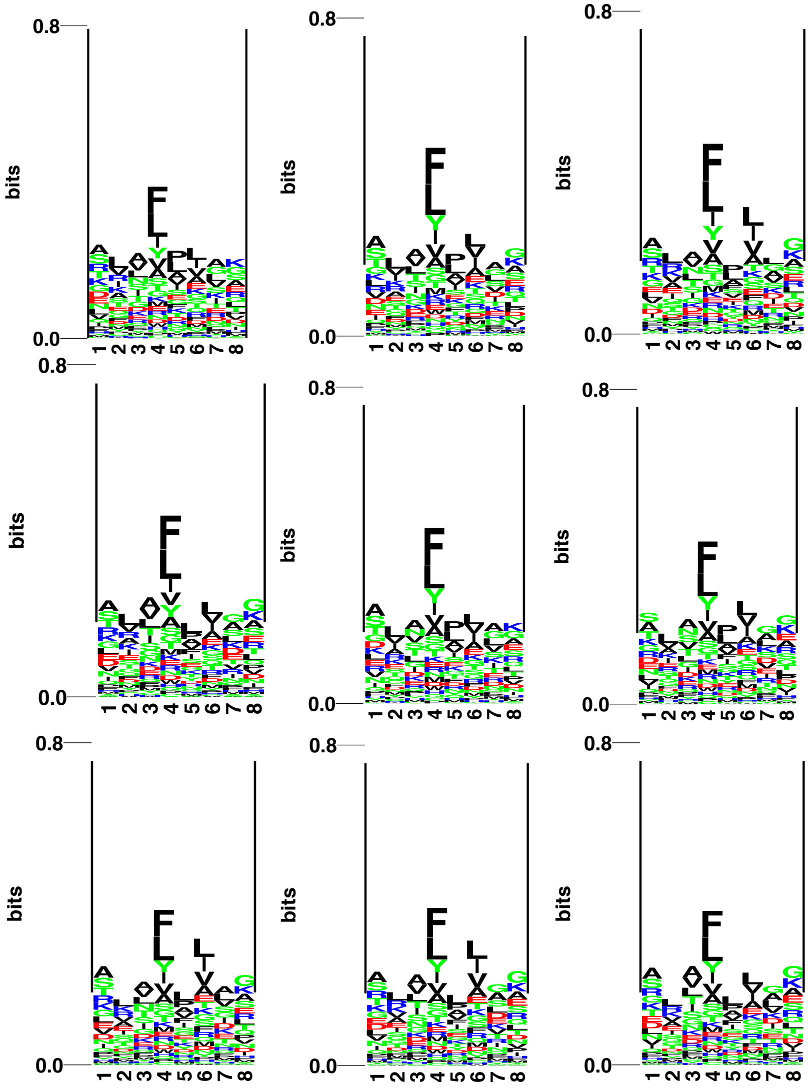


Figure 5.5. Sequence logos for each of the nine 9-fold average matrices, where the matrix for test set 1 is in the top left corner, and for test set 9 in the bottom right corner.

Table 5.3. 9-fold cross-validated performances

Test set	9-fold avemat AUC
1	0.976
2	0.784
3	0.815
4	0.997
5	0.880
6	0.848
7	0.974
8	0.968
9	0.997
Average	**0.915**

The table is listing the 9-fold avemat performances.

Leu at position 4. Logo number 3, 7 and 9 also seem to have a higher preference for Leu at position 6. The reason for the apparent loss of motifs from the individual substrate matrices, compared to these, is the fact that the sequence logos in Figure 5.5 are the result of the average scores using the 9-fold cross validation scheme. If for example Pro scores high for some of the individual substrate sequences, where it is very abundant, it will on the other hand have a very negative score in the substrate matrices where Pro is not appearing, and the average result will be a low score for Pro. The average matrix thereby reflects this and Pro is not seen as an abundant amino acid in the logo.

All test sets were evaluated as can be seen in Table 5.3. For all test sets, the left out substrate peptides together with the negative set, were used to calculate the performance measured in AUC. Using 9-fold avemat an average AUC of 0.915 was reached.

In order to find peptides from the negative dataset which were predicted with a positive score, a final average matrix was created. This matrix was an average of the nine matrices, which were created for each individual substrate data set. All the peptides in the negative dataset was scored using the average matrix. The final score is the sum of the log-odds scores for the amino acids at each of the eight positions in the peptide.

The top10 and bottom10 ranking peptides for both the positive and the negative dataset are listed in Table 5.4. The table indicates that some of the peptides from the negative dataset have the potential to be possible cleavage sites due to their positive score. They do not score as high as the highest scoring positives, but they do seem to share some of the same characteristics as the cleavable substrates, as indicated by the positive score. It is important to note, that since there is no defined threshold for what is cleaved and what is not cleaved, peptides with a positive score is not necessarily cleavable

Table 5.4. Top10 and bottom10 predictions for the positive and negative datasets

Sequence	Score	Sequence	Score
Top10 Negatives		***Bottom10 Negatives***	
SQIYPGIK	3.054	LHPDKWTV	-12.537
TPVFAIKK	3.011	IWGKTPKF	-12.602
KELYPLTS	2.688	YDPSKDLI	-12.817
TEVIPLTE	2.486	LDVGDAYF	-12.819
AREFSSEQ	2.349	PGMDGPKV	-12.939
TQDFWEVQ	1.474	IGGIGGFI	-13.014
ETAYFLLK	1.356	IGPENPYN	-13.266
VKQWPLTE	1.338	WIPEWEFV	-13.442
AEVIPAET	1.163	KGRPGNFL	-13.679
KFKLPIQK	1.077	LNFPISPI	-13.816
Top10 Positives		***Bottom10 Positives***	
SQNFPIVQ	8.502	NCDGLEAQ	-3.974
TLNFPISQ	8.488	RVLAEAMS	-4.224
SRNFPIVQ	8.441	ERQANFLG	-4.553
SQNFPIIQ	8.366	DRQANFLG	-4.579
SQNFPLVQ	8.247	DSGGLRAQ	-4.788
SKNFPIVQ	8.061	ACGGLGAQ	-4.856
SLAFPQGK	7.981	ERRANFLG	-5.048
ARAFPQGK	7.959	RQANFGKF	-5.319
SLNFPISP	7.885	RQANFGEF	-5.404
SQNYPIAQ	7.768	EGQANFLG	-6.863

The table lists the top10 and bottom10 predictions from the positive and negative dataset and their predicted score.

substrates, but there is a higher chance for it. This applies both to peptides from the positive and the negative dataset. It is also seen from the table that it is possible for peptides from the positive dataset to have a negative score. This indicates that these variants are more different from the characteristics of the average substrate. It may very well still be a cleavable site, the reason could be that it is from a mutated protease which has changed its specificity. Whether or not both the highest ranking peptides from the negative dataset is actually cleavable sites is yet to be verified in the lab.

5.4 Future perspectives

Unfortunately there were a lot of time constraits during the period of this project. It could have been interesting to further pursue the so far promising results. A high performance value of 0.915 measured in AUC was reached; using the 9-fold cross validated average matrices.

If time permitted, it could have been interesting to further evaluate the method using peptides which have been verified in the lab. Such a dataset was obtained from the lab of Jan Komorowski (Kontijevskis et al. (2007)) close to the final delivery date of this thesis. The dataset contains 16 years of HIV-1 Protease and its substrate interactions research with a total of 1625 substrates (374 are cleavable and 1251 noncleavable). All substrates have been experimentally verified in the lab for cleavage with the HIV-1 wild-type.

Steric hindrance could be a reason for the ability to cleave or not cleave a substrate. It has been suggested by among others Chaudhury and Gray (2009) that an underlying structural mechanism could be important for the specificity. It would therefore have been interesting to examine the secondary structure conformation in and around the naturally occurring substrate sites, and also examining the positive scoring peptides from the negative dataset. It may very well be, that the peptides are not being cleaved due to a mechanism involving steric hindrance.

Moreover, 10-20 of the highest ranking peptides from the negative dataset will be validated for cleavability in the lab of Martin Willemoës. It will be exciting to hear what the results of the experiments are.

Chapter 6

Discontinuous B-cell epitope prediction

The following chapter describes the work performed in an attempt to improve an already existing in-house tool, DiscoTope, which is a method for prediction of discontinuous B-cell epitopes from a proteins three-dimensional structure. The method was published in 2006 by Andersen et al. (Haste Andersen et al. (2006)) and is also available as a web-server. The motivation for starting the project was the expectation that DiscoTope could be further improved by implementation of Half Sphere Exposure measures.

6.1 Background

Prediction of B-cell epitopes has the potential to help identifying epitopes in proteins for further selection and use in peptide-based vaccines. Most of the B-cell epitope prediction methods have been focused on the area of linear epitopes. Unfortunately only $\sim$ 10% of B-cell epitopes are continuous in the sequence (Van Regenmortel and Pellequer (1994)) and 90% are therefore discontinuous or conformational epitopes. Conformational epitopes are distantly separated in sequence, but brought together in proximity in the three-dimensional structure. These epitopes are furthermore more difficult to identify, due to the nature of the way they have to be analysed. For the linear epitopes, sequences are cut into peptides and the antibody binding affinity is measured. For the conformational epitopes one needs to measure the binding of a whole protein to the antibody, thereby making it more difficult. The most accurate annotation method for conformational epitopes is X-ray crystallisation of the entire antibody-antigen complex. In the work presented here, epitope identification has been performed by use of antibody-antigen coordinate datafiles obtained from PDB. Antigen amino acids having atoms within a 4Å distance from antibody atoms were defined as epitope residues.

The characteristics of B-cell epitopes have previously been investigated and it was discovered that generally they are charged, hydrophilic, surface exposed and located in flexible regions (Ansari and Raghava (2010)). Furthermore, a study by Pellequer et al. (1991) showed that 50% of the linear B-cell epitopes in a small dataset of 11 proteins were located in a turn region.

Two methods using surface accessibility and protein structure for prediction of discontinuous B-cell epitopes are the Conformational Epitope Predictor (CEP) web-server (Kulkarni-Kale et al. (2005)) and an in-silico method by Batori et al. (2006). DiscoTope (Haste Andersen et al. (2006)) uses a combination of amino acid statistics, spatial information and surface exposure information. Rapberger et al. (2007) describes a method, where surface accessibility, shape complementarity and binding energies are combined. Pepito (Sweredoski and Baldi (2008)) further improved the prediction performance by using a multiple distance threshold and Half Sphere Exposure measures. Ten days later Song et al. (2008) published a web-server, HSEpred, which also had Half Sphere Exposure values incorporated. The limitation of these methods lies in that they all require the tertiary structure of the antigen. Lately a method, CBTOPE, that only requires the primary sequence for prediction of conformational epitopes has been developed (Ansari and Raghava (2010)). Using a SVM based model with amino acid physico-chemical patterns, composition profiles and sparse encoding a performance of MCC 0.73 was obtained on a dataset of 187 protein chains.

Recently a study of 75 experimentally determined antigen-antibody structures with predicted epitope residues from DiscoTope (Haste Andersen et al. (2006)) was carried out. It was found that there was an over-representation of β-turns with a factor of 2 in the discontinuous B-cell epitopes (Petersen et al. (2010)). This indicates that the prediction of β-turn occurrence could possibly improve the prediction of epitopes. β-turn and coil formation have previously been used to predict linear epitopes (Alix (1999)).

6.1.1 Half Sphere Exposure

Half Sphere Exposure (HSE) is a new measure of solvent exposure introduced by Thomas Hamelryck (Hamelryck (2005)). HSE is different from other solvent exposure measures in the way that it splits the number of contacts into two measures, HSE-up and HSE-down. Compared with other solvent exposure measures, HSE has shown to be superior in terms of correlation with protein stability (Hamelryck (2005)). The first step of calculation of the two measures is to identify all C_α-C_α contacts within a sphere with a predefined radius from the residues C_α-atom. A plane perpendicular to the C_α-C_β vector is constructed splitting the sphere in two halves. The upwards direction of the C_α-C_β vector is called HSE-up, and the downwards direction is called HSE-down. An illustration of this can be seen in Figure 6.1.

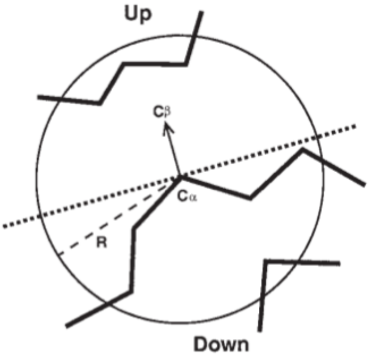


Figure 6.1. Illustration of Half Sphere Exposure (HSE). The dotted line represents the plane dividing the sphere around the C_α-atom in two halves, HSE-up and HSE-down. Here the measures, or C_α-C_α counts, for HSE-up and HSE-down are 3 and 5, respectively. Picture courtesy of Hamelryck (2005).

6.2 Materials and Methods

6.2.1 Data set

For the training and optimization of the method, five datasets were downloaded from the supplementary section of the DiscoTope web-server (Haste Andersen et al. (2006)). The DiscoTope datasets consist of 75 experimentally determined protein antigen-antibody structures with previously assigned epitope residues. These five datasets are further called 'disco4'. One new datasets was created where epitope residues were defined as antigen amino acids having atoms within a distance of 4Å (dist4 dataset) from antibody atoms. The dataset consists of the same protein chains split into five test sets as previously described (Haste Andersen et al. (2006)). All performance measures are therefore reported as five-fold cross-validation measures. The dist4 dataset consist of 14,448 amino acids with 1,204 amino acids assigned as epitope residues.

6.2.2 Propensity scores

Propensity Scores (PS) are composed of values describing intrinsic features for each of the 20 amino acids. PS were calculated for the dist4 dataset, using a 5-fold cross validation scheme (see Section 1.2). This implies that four of the five datasets (training sets) were used for the calculation of the epitope log-odds ratios, 5 matrices were created in total for each dataset. The peptides were produced by sliding a window of the size 9 though the sequences in the training set. Peptides were thereafter sorted into two groups, epitope or non-epitope, depending on the residue in the middle of the window. PSSM's or wright matrices were then calculated from the peptides in each group. To correct for possible sampling biases, peptides were clustered using a 62% identity threshold as described by (Henikoff and Henikoff (1994)). It was done in order to down-weight the peptides, which showed a high degree of similarity. A weight of 1/N was hereafter assigned to all peptides in the clusters, where N is the number of peptides in the cluster. Therefore none of the peptides was removed, but each of them had a lower weight in the statistics, when the matrix was generated. To make sure all amino acids were represented in the dataset, pseudocount correction was used (Altschul et al. (1997)). Final log-odds ratios were generated for both datasets by subtracting the weights from the fifth position in the epitope matrix with the weights from the fifth position in the non-epitope matrix. This was performed for all 20 amino acids and the final scores were used as propensity scores.

For the disco4 dataset surface exposure dependent propensity scores were also calculated. All amino acids were assigned as exposed/ non-exposed using DSSP and the following exposure levels: 0%, 10%, 15%, 25% and 30%. Only epitopes assigned as being exposed based on the exposure level, were included in the epitope group when the peptides were sorted for calculation of the epitope/ non-epitope matrices in the final calculation of the propensity scores.

6.2.3 Prediction

For each residue *r* in the target protein chain, a score, S(r) is calculated by using Equation 6.1.

$$S(r) = \sum_{d} PS(r) - \alpha \cdot HSEup(r,d) - \beta \cdot HSEdown(r,d) \tag{6.1}$$

HSEup(r,d) is the HSE-up values for residue *r* using the distance *d* for the sphere. Similarly *HSEdown(r,d)* is the HSE-down values for residue *r* using the distance *d* for the sphere. α and β are weights on HSE-up and HSE-down and *PS(r)* is the propensity score for amino acid *r*.

6.3 Results and Discussion

PRedictions were performed for the disco4 dataset using the in-house version of DiscoTope with all surface exposure dependent propensity scores. The in-house version was used (with no HSE implemented) in order to find the optimal surface exposure propensity scores.

As shown in Table 6.1 the highest performance was reached with the threshold for surface exposure measure of 25%. This correlates well with the fact that a surface exposure threshold of 25% is most often used for classification of an exposed residue, and that epitopes are in exposed or protruding regions. Using a threshold of 25% the performance reaches an AUC of 0.715, while an AUC of 0.700 is reached when no threshold is used.

Table 6.1. Performance measures for surface exposure dependent propensity scores

Surface threshold	**Average AUC**
0%	0.700
10%	0.702
15%	0.707
25%	0.715
30%	0.709

The table lists the performance measured in AUC for the different surface exposure dependent propensity scores.

Pepito (Sweredoski and Baldi (2008)) was published as one of the first methods using Half Sphere Exposure. Sweredoski and Baldi (2008) is directly comparing their performance with the performance obtained from DiscoTope. In order to make a comparison between performances obtained in Pepito and this work possible, trainings were performed using the same settings with the dist4 dataset. The result of these trainings can be seen in Table 6.2.

Table 6.2. Performances for the different setups used in the trainings

Training #	α	β	**d**	**Propensity Scores**	**AUC**
dist4 dataset:					
1	0.5	0.25	8, 10, 12, 14, 16	dist4	0.710
2	0.5	0.25	10	dist4	0.698
3	0.5	0.25	8, 10, 12, 14, 16	DiscoTope Paper	0.718
4	0.5	0.25	10	DiscoTope Paper	0.708
5	0.5	0.25	8, 10, 12, 14, 16	dist4 surface exp	0.700
6	0.5	0.25	10	dist4 surface exp	0.697
7	0.5	0.5	10	DiscoTope Paper	0.704
disco4 dataset:					
8	0.5	0.5	10	disco4 surface exp	0.715

The table lists the different setups and performances obtained, measured in AUC. 'DiscoTope paper' refers to the Propensity Scores (PS) published in (Haste Andersen et al. (2006)), 'dist4' is PS calculated from the dist4 dataset, 'dist4 surface exp' is PS calculated from epitopes, which are more than 25% exposed, 'disco4 surface exp' is the before mentioned PS used on the disco4 dataset.

Table 6.2 lists the obtained performances from the different trainings with the dist4 dataset. Training 7 is using settings which are comparable to DiscoTope. DiscoTope uses the full sphere (α and β both of 0.5) with a sphere radius of 10Å. The obtained performance, AUC 0.704, is slightly lower than the published performance for DiscoTope, AUC 0.711. Due to unclear reasons, it was not possible to reach the exact same performance as in DiscoTope, even though the same settings were used. The datasets do differ by one epitope assigned amino acid, but the method for calculating the final score is the same. The significance of the difference has not been tested. Nevertheless, it was decided to continue with the dist4 dataset and the newly calculated propensity scores.

Dist4 was used for trainings with HSE-up and HSE-down measures. In some of the trainings only one distance measure for the half spheres were used, but in others multiple distance measures were used as can be seen in Table 6.2.

The highest performance obtained was for training 3, which uses the published DiscoTope PS, a weight of 0.5 on HSE-up, a weight of 0.25 on HSE-down and multiple distances of 8, 10, 12, 14 and 16 for the calculation of the HSE measures. The obtained performance was an AUC of 0.718, which is clearly higher than the performance obtained when only using one distance was used, which gave an AUC of 0.708 (training 4). These performances are higher than the cross-validated dist4 performances (training 1/2), because of over-training due to the reason that the published PS were used. The finding that multiple distance thresholds improve the performance was first published by Sweredoski and Baldi (2008). The same finding was also observed here for all training examples.

Using the surface exposure dependent PS (training 5/6) did not improve the performance compared to using the DiscoTope PS (training 3/4), which indicates that HSE itself is a good measure for explaining the surface exposure of the epitope residues.

Pepito have a published performance of AUC 0.754 on the DiscoTope dataset, which is 0.036 higher than what we obtained when using the same dataset and the same settings. The reason for this is unknown. It is therefore not possible to directly compare the performances obtained in this work with the ones obtained by Sweredoski and Baldi (2008).

The performance reached in training 1, an AUC of 0.710, which was obtained using PS calculated for dist4 and multiple distance threshold, is comparable to the performance obtained by DiscoTope. We did obtain a slightly higher performance than DiscoTope when using DiscoTope PS, a weight of 0.5 on HSE-up, a weight of 0.25 on HSE-down and a multiple distance thresholds of 8, 10, 12, 14 and 16, which gave rise to an AUC of 0.718 (training 3).

6.4 Future perspectives

If time permitted, a more in-depth and thorough analysis could be done to understand the differences between the performance of DiscoTope and that obtained in the current work. It would also be interesting to see if one could further optimize the α and β values to see if this could improve the performance, since it was not tried in the work of Sweredoski and Baldi

(2008). A study showed that there was an over representation of β-turns in the B-cell epitopes in the DiscoTope dataset with a factor of 2. Therefore use of β-turn assignment to improve the prediction could also have been interesting to examine.

Chapter 7

Summary & perspectives

The field of bioinformatics has exploded within the last 15-20 years and the amount of data available in public databases is enormous. GenBank, which is a genetic sequence database, now contains more than 132 million sequences and the Protein Data Bank contains more than 62,200 X-ray solved structures. When the amount of publicly available DNA sequences is compared with the amount of experimentally verified proteins, a huge over-representation of DNA sequences can be seen. In order for the academic and industrial research community to gain use of this plethora of information, one has to rely on structure predictions rather than investing time consuming experimentally determined structure data.

The two major goals for performing the research presented in this thesis was: 1. To develop a method for the successful prediction of surface accessibility of amino acids in an amino acid sequence. 2. To develop a method for prediction of β-turns and the specific β-turn types. Both goals were successfully fulfilled and resulted in two published peer-reviewed papers. Furthermore the two methods are available online as web-servers: NetSurfP and NetTurnP, respectively.

Chapter 1 gives a general introduction to the tools and performance measures used for the projects throughout the thesis. A more in depth introduction for the specific projects can be found in their respective chapters.

Chapter 2 deals with the prediction of protein surface accessibility. The method NetSurfP produced a state-of-the-art web-server, which is freely available online. For academic users it is also possible to download an executable version of NetSurfP. Since its launch in August 2009, NetSurfP has become quite popular with more than 40 citations in peer-reviewed journals with more than 161,500 computed amino acid sequences being analyzed. NetSurfP predicts the surface accessibility of amino acids in a protein sequence and both the buried/exposed classification and its relative surface area are reported to the user. Simultaneously, the method also predicts the reliability for each prediction in the form of a Z-score. Furthermore, the secondary structure prediction is also presented to the user. NetSurfP obtained an accuracy of 79% correctly predicted residues at the classification networks, buried/exposed, and a Pearsons Correlation Coefficient of 0.72 at the real value networks.

Chapter 3 describes the work of NetTurnP, which was the second main goal of this thesis. β-turns are described as the most common type of non-repetitive structures as they constitute on average 25% of the amino acids in proteins. Moreover, they were found to be overrepresented in B-cell epitopes with a factor of 2. Predicted secondary structure and surface accessibility from NetSurfP were used to improve the prediction of β-turns in the NetTurnP method. NetTurnP achieved a Matthews Correlation Coefficient of 0.50, which is the highest performance reported on a two-class prediction of β-turn or not-β-turn. Furthermore, NetTurnP showed improved performance on some of the specific β-turn types.

Chapter 4 deals with a caching project, which was launched in order to improve the speed of NetSurfP and NetTurnP. Both methods are dependent on the creation of Position Specific Scoring Matrices (PSSM), which requires the query sequence to be blasted for four iterations against a non-redundant database. The first step of the caching, which involved pre-calculating more than 500,000 sequences as PSSM, improved the prediction time for 1,000 random human sequences from 77 hours and 20 minutes to 9 hours and 42 minutes. Due to time limitation the last step involving a more obvious direct look-up for NetSurfP output, was not implemented since it requires most of the NetSurfP code to be rewritten. A benchmark showed that the total calculation time for the same 1,000 human sequences could be reduced to 14 minutes using the direct look-up. This shall be implemented at a later stage.

Chapter 5 describes the work performed in Martin Willemoës' lab at the Department of Biology at Copenhagen University. The project aims were to predict the HIV-1 Protease specificity, to identify variants of known substrates (small peptides of eight amino acids) and to identify new cleavable substrates. The HIV-1 protease is completely essential to the lifecycle of the HIV virus, and a non-functional protease results in immature and non-infectious HIV virus particles. The HIV protease is therefore an important target for the rational design of drugs against HIV. In the process of examining the HIV protease specificity different variants for nine out of twelve known substrates were identified. Furthermore, as a result a cross-validated AUC of 0.915 was obtained, and new possible substrates were selected for further verification in the lab. Unfortunately, due to time constraints the promising results were not further pursued. It would have been interesting to further evaluate the method using a dataset consisting of experimentally verified substrates (374 cleavable and 1251 non-cleavable), which was obtained from Jan Komorowski (Kontijevskis et al. (2007)). Examining the secondary structure conformation in and around non-cleavable sites, which were predicted to be possible cleavable sites, could have been an interesting approach. This could further elucidate the theory saying that an underlying structural mechanism is the reason for the ability to cleave or not cleave a substrate.

The last part of the thesis, presented in chapter 6, dealt with the prediction of discontinuous B-cell epitopes. Half Sphere Exposure was integrated as a new measure for describing the exposure of epitope residues in an already existing tool, DiscoTope. Due to many setbacks stemming from replicating the previously obtained results (from DiscoTope), there was not enough time to allow the full completion of this project. Due to unknown reasons, when predictions using the same settings and dataset as Pepito (Sweredoski and Baldi (2008)) was performed, a lower performance was obtained. However, an improvement of the performance compared to the published DiscoTope performance was reached. Further optimization of the weights on the Half Sphere Exposure measures and integration of other modules are expected to further improve the performance of the method.

Bibliography

Adamczak R., Porollo A., and Meller J. (2004). Accurate Prediction of Solvent Accessibility Using Neural NetworksBased Regression. *Proteins: Structure, Function, and Bioinformatics*, 56:753–767. 30

Ahmad S., Gromiha M.M., and Sarai A. (2003). Real value prediction of solvent accessibility from amino acid sequence. *Proteins: Structure, Function, and Bioinformatics*, 50(4):629–635. 21, 22, 29, 30, 32, 34, 37, 39, 111

Alix A.J. (1999). Predictive estimation of protein linear epitopes by using the program PEOPLE. *Vaccine*, 18(3-4):311–4. 49, 58, 92

Altschul S.F., Madden T.L., Schaffer A.A., Zhang J., Zhang Z., et al. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389. 7, 38, 69, 84, 94

Ansari H.R. and Raghava G.P. (2010). Identification of conformational B-cell Epitopes in an antigen from its primary sequence. *Immunome Res*, 6:6. 92

Armitage P., Berry G., Matthews J.N.S., and Corporation E. (2002). *Statistical Methods in Medical Research*. Wiley Online Library. 64

Barton G. (2007). Jpred Distribution material. 38

Batori V., Friis E.P., Nielsen H., and Roggen E.L. (2006). An in silico method using an epitope motif database for predicting the location of antigenic determinants on proteins in a structural context. *J Mol Recognit*, 19(1):21–9. 92

Benson D.A., Karsch-Mizrachi I., Lipman D.J., Ostell J., and Wheeler D.L. (2005). GenBank. *Nucleic Acids Res*, 33(Database issue):D34–8. 3

Berman H.M., Westbrook J., Feng Z., Gilliland G., Bhat T.N., et al. (2000). The Protein Data Bank. *Nucl. Acids Res.*, 28(1):235–242. 3, 38, 46, 66

Blicher T., Gupta R., Wesolowska A., Jensen L.J., and Brunak S. (2010). Protein annotation in the era of personal genomics. *Curr Opin Struct Biol*, 20(3):335–41. 6, 75

Cai Y.D., Yu H., and Chou K.C. (1998). Artificial neural network method for predicting HIV protease cleavage sites in protein. *J Protein Chem*, 17(7):607–15. 81

Cai Y.D.D., Liu X.J.J., Xu X.B.B., and Chou K.C.C. (2002). Support Vector Machines for predicting HIV protease cleavage sites in protein. *J Comput Chem*, 23(2):267–74. 81

Carugo O. (2000). Predicting residue solvent accessibility from protein sequence by considering the sequence environment. *Protein Eng*, 13(9):607–609. 30

Chaudhury S. and Gray J.J. (2009). Identification of Structural Mechanisms of HIV-1 Protease Specificity Using Computational Peptide Docking: Implications for Drug Resistance. *Structure*, 17(12):1636–1648. 90

Cheng J., Randall A.Z., Sweredoski M.J., and Baldi P. (2005). SCRATCH: a protein structure and structural feature prediction server. *Nucleic Acids Res*, 33(Web Server issue):W72–6. 22, 30

Chothia C. (1976). The nature of the accessible and buried surfaces in proteins. *Journal of molecular biology*, 105(1):1–12. 21, 29, 39, 111

Chou P.Y. and Fasman G.D. (1974). Conformational parameters for amino acids in helical, -sheet, and random coil regions calculated from proteins. *Biochemistry*, 13(2):211–222. 56

Chou P.Y. and Fasman G.D. (1979). Prediction of beta-turns. *Biophysical Journal*, 26. 56

Clemente J.C., Coman R.M., Thiaville M.M., Janka L.K., Jeung J.A., et al. (2006). Analysis of HIV-1 CRF_01 A/E Protease Inhibitor Resistance: Structural Determinants for Maintaining Sensitivity and Developing Resistance to Atazanavir. *Biochemistry*, 45(17):5468–5477. 80

Connolly M.L. (1983). Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16(5):548–558. 21, 29

Crooks G.E., Hon G., Chandonia J.M.M., and Brenner S.E. (2004). WebLogo: a sequence logo generator. *Genome Res*, 14(6):1188–90. 16

Cuff J.A. and Barton G.J. (1999). Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 34(4):508–519. 38

DeLano W.L.D. (2002). An open-source molecular graphic tool. *CCP4 newsletter of Protein Crystallography*, 40. xvi, 67

Dor O. and Zhou Y. (2007a). Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training. *Protein: Structure, Function, and Bioinformatics*, 66(4):838–845. 21, 22, 30, 32, 37

Dor O. and Zhou Y. (2007b). Real-SPINE: an integrated system of neural networks for real-value prediction of protein structural properties. *Proteins: Structure, Function, and Bioinformatics*, 68(1):76–81. 22, 30, 32, 33, 34, 37

Dyrl Bendtsen J., Nielsen H., von Heijne G., and Brunak S. (2004). Improved Prediction of Signal Peptides: SignalP 3.0. *Journal of Molecular Biology*, 340(4):783–795. 4

Ettmayer P., France D., Gounarides J., Jarosinski M., Martin M.S., et al. (1999). Structural and conformational requirements for high-affinity binding to the SH2 domain of Grb2(1). *J Med Chem*, 42(6):971–80. 48, 58

Faraggi E., Xue B., and Zhou Y. (2009). Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network. *Proteins: Structure, Function, and Bioinformatics*, 74(4):847–856. 30, 37

Fuchs P.F. and Alix A.J. (2005). High accuracy prediction of beta-turns and their types using propensities and multiple alignments. *Proteins: Structure, Function, and Bioinformatics*, 59(4):828–839. 48, 56, 57, 61, 63

Garg A., Kaur H., and Raghava G.P.S. (2005). Real value prediction of solvent accessibility in proteins using multiple sequence alignment and secondary structure. *Proteins: Structure, Function, and Bioinformatics*, 61(2):318–324. 30

Garnier J., Osguthorpe D.J., and Robson B. (1978). Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *Journal of Molecular Biology*, 120(1):97–120. 56

Garnier J. and Robson B. (1989). Prediction of protein structure and the principles of protein conformation. *Plenum Press, New York*, pages 417–465. 56

Gorodkin J., Heyer L.J., Brunak S., and Stormo G.D. (1997). Displaying the information contents of structural RNA alignments: the structure logos. *Comput Appl Biosci*, 13(6):583–6. 16

Gupta R.G., Jung E.J., and Brunak S.B. (2004). Prediction of N-glycosylation sites in human protein. *In preparation, 2004*. 4

Guruprasad K. and Rajkumar S. (2000). Beta-and gamma-turns in proteins revisited: a new set of amino acid turn-type dependent positional preferences and potentials. *J Biosci*, 25(2):143–156. 65

Hahn B.H., Shaw G.M., Arya S.K., Popovic M., Gallo R.C., et al. (1984). Molecular cloning and characterization of the HTLV-III virus associated with AIDS. *Nature*, 312(5990):166. 82

Hamelryck T. (2005). An amino acid has two sides: a new 2D measure provides a different view of solvent exposure. *Proteins*, 59(1):38–48. xvii, 92, 93

Haste Andersen P., Nielsen M., and Lund O. (2006). Prediction of residues in discontinuous B-cell epitopes using protein 3D structures. *Protein Science*, 15(11):2558–2567. 21, 30, 49, 65, 91, 92, 93, 95

Henikoff S. and Henikoff J.G. (1992). Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89(22):10915. 7

Henikoff S. and Henikoff J.G. (1994). Position-based sequence weights. *J Mol Biol*, 243(4):574–8. 84, 94

Hobohm U., Scharf M., Schneider R., and Sander C. (1992). Selection of representative protein data sets. *Protein Science*, 1(3):409–417. 41, 66

Hu X. and Li Q. (2008). Using support vector machine to predict β and γturns in proteins. *Journal of Computational Chemistry*, 29(12):1867–1875. 57, 61

Huang Y., Niu B., Gao Y., Fu L., and Li W. (2010). CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, 26(5):680. 7

Hutchinson E.G. and Thornton J.M. (1994). A revised set of potentials for beta-turn formation in proteins. *Protein Sci*, 3(12):2207–2216. 46, 48, 56

Hutchinson E.G. and Thornton J.M. (1996). PROMOTIFa program to identify and analyze structural motifs in proteins. *Protein Sci*, 5(2):212–220. 46, 68, 72, 118

Illergård K., Ardell D.H., and Elofsson A. (2009). Structure is three to ten times more conserved than sequence–a study of structural response in protein cores. *Proteins*, 77(3):499–508. 7

James M.W. and Poet R.P. (1987). Loops, bulges, turns and hairpins in proteins. *Trends in Biochemical Sciences*, 12:189–192. 56

Jones S. and Thornton J.M. (1997a). Analysis of protein-protein interaction sites using surface patches. *J Mol Biol*, 272(1):121–132. 21, 30

Jones S. and Thornton J.M. (1997b). Prediction of protein-protein interaction sites using patch analysis. *Journal of molecular biology*, 272(1):133–143. 21, 30

Kabsch W. and Sander C. (1983). Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637. 24, 38, 39, 41, 45

Katoh K., Misawa K., Kuma K.i., and Miyata T. (2002). MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res*, 30(14):3059–66. 83

Kaur H. and Raghava G.P.S. (2002). An evaluation of beta-turn prediction methods. *Bioinformatics*, 18(11):1508–1514. 57

Kaur H. and Raghava G.P.S. (2004). A neural network method for prediction of beta-turn types in proteins using evolutionary information. *Bioinformatics*. 57, 63

Kaur H. and Raghava G.P.S.R. (2003). Prediction of beta-turns in proteins from multiple alignment using neural network. *Protein Science*, 12(3):627–634. 57, 61, 63

Kim S. (2004). Protein beta-turn prediction using nearest-neighbor method. *Bioinformatics*, 20(1):40. 57, 61

Kirschner A. and Frishman D. (2008). Prediction of β-turns and β-turn types by a novel bidirectional Elman-type recurrent neural network with multiple output layers (MOLEBRNN). *Gene*, 422(1-2):22–29. 61, 63

Kontijevskis A., Wikberg J.E., and Komorowski J. (2007). Computational proteomics analysis of HIV-1 protease interactome. *Proteins: Structure, Function, and Bioinformatics*, 68(1):305–312. 85, 90, 100

Kountouris P. and Hirst J. (2010). Predicting beta-turns and their types using predicted backbone dihedral angles and secondary structures. *BMC Bioinformatics*, 11(1):407. 61, 63

Krchnak V., Mach O., and Mal A. (1987). Computer prediction of potential immunogenic determinants from protein amino acid sequence. *Analytical Biochemistry*, 165(1):200–207. 58

Krchnak V., Mach O., and Mal A. (1989). Computer prediction of B-cell determinants from protein amino acid sequences based on incidence of [beta] turns. *Methods in Enzymology*, 178:586–611. 58

Kulkarni-Kale U., Bhosle S., and Kolaskar A.S. (2005). CEP: a conformational epitope prediction server. *Nucleic Acids Res*, 33(Web Server issue):W168–71. 92

Larkin M.A., Blackshields G., Brown N.P., Chenna R., McGettigan P.A., et al. (2007). Clustal W and Clustal X version 2.0. *Bioinformatics*, 23(21):2947–8. 83

Li W., Jaroszewski L., and Godzik A. (2001). Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, 17(3):282. 38, 69

Liu L., Fang Y., Li M., and Wang C. (2009). Prediction of Beta-Turn in Protein Using E-SSpred and Support Vector Machine. *The Protein Journal*, 28(3-4):175–181. 57, 61

Lund O., Frimand K., Gorodkin J., Bohr H., Bohr J., et al. (1997). Protein distance constraints predicted by neural networks and probability density functions. *Protein Engineering Design and Selection*, 10(11):1241. 66

Lund O., Nielsen M., Lundegaard C., Kesmir C., and Brunak S. (2005). *Immunological Bioinformatics*. The MIT Press, Cambridge, Massachusetts, London, England. 19, 39, 69, 71, 84

Lundegaard C., Lamberth K., Harndahl M., Buus S., Lund O., et al. (2008). NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8-11. *Nucleic Acids Res*, 36(Web Server issue):W509–12. 4

Lundegaard C., Lund O., Kesmir C., Brunak S., and Nielsen M. (2007). Modeling the adaptive immune system: predictions and simulations. *Bioinformatics*, 23(24):3265–75. 29

Matthews B.W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim Biophys Acta*, 405(2):442–451. 18, 70

McGregor M.J., Flores T.P., and Sternberg M.J. (1989). Prediction of beta-turns in proteins using neural networks. *Protein Eng*, 2(7):521–6. 57

McGuffin L.J., Bryson K., and Jones D.T. (2000). The PSIPRED protein structure prediction server. *Bioinformatics*, 16(4):404–405. 57

Mooney S. (2005). Bioinformatics approaches and resources for single nucleotide polymorphism functional analysis. *Briefings in bioinformatics*, 6(1):44. 22, 30

Nalam M.N.L., Ali A., Altman M.D., Reddy G.S.K.K., Chellappan S., et al. (2010). Evaluating the Substrate-Envelope Hypothesis: Structural Analysis of Novel HIV-1 Protease Inhibitors Designed To Be Robust against Drug Resistance. *Journal of Virology*, 84(10):5368–5378. 80

Nguyen M.N. and Rajapakse J.C. (2006). Two-stage support vector regression approach for predicting accessible surface areas of amino acids. *Proteins: Structure, Function, and Bioinformatics*, 63(3):542–550. 32, 33

Nielsen M., Lundegaard C., Lund O., and Petersen T.N. (2010). CPHmodels-3.0 - remote homology modeling using structure-guided sequence profiles. *Nucl. Acids Res.*, 38:576–581. 6

Panchenko A.R. (2004). Prediction of functional sites by analysis of sequence and structure conservation. *Protein Science*, 13(4):884–892. 22, 30

Pellequer J.L.P., Westhof E.W., and RegenMortel M.H.V.V.R. (1991). Predicting location of continuous epitopes in proteins from their primary structures. *Methods in Enzymology*, 203:176–201. 49, 57, 92

Perez M.A.S., Fernandes P.A., and Ramos M.J. (2010). Substrate Recognition in HIV-1 Protease: A Computational Study. *The Journal of Physical Chemistry B*, 114(7):2525–2532. xvii, 81, 82

Perryman A.L., Lin J.H.H., and McCammon J.A. (2004). HIV-1 protease molecular dynamics of a wild-type and of the V82F/I84V mutant: possible contributions to drug resistance and a potential new target site for drugs. *Protein Sci*, 13(4):1108–23. xvii, 80

Petersen B., Lundegaard C., and Petersen T.N. (2010). NetTurnP - Neural Network Prediction of Beta-turns by Use of Evolutionary Information and Predicted Protein Sequence Features. *PLoS One*, 5(11):e15079. 6, 92

Petersen B., Petersen T., Andersen P., Nielsen M., and Lundegaard C. (2009). A generic method for assignment of reliability scores applied to solvent accessibility predictions. *BMC Structural Biology*, 9(1):51. xv, xvi, xvii, 59, 60, 66, 69, 70, 71, 72, 115, 116, 119

Petersen T.N., Lundegaard C., Nielsen M., Bohr H., Bohr J., et al. (2000). Prediction of protein secondary structure at 80% accuracy. *Proteins: Structure, Function, and Genetics*, 41(1):17–20. 41, 70

Pham T.H., Satou K., and Ho T.B. (2003). Prediction and analysis of beta-turns in proteins by support vector machine. *Genome Informatics. International Conference on Genome Informatics*, 14:196–205. 61

Pollastri G., Baldi P., Fariselli P., and Casadio R. (2002). Prediction of coordination number and relative solvent accessibility in proteins. *Proteins: Structure, Function, and Bioinformatics*, 47(2):142–153. 22, 30

Pollastri G., Martin A.J., Mooney C., and Vullo A. (2007). Accurate prediction of protein secondary structure and solvent accessibility by consensus combiners of sequence and structure information. *BMC bioinformatics*, 8(1):201. 30

Press W.H., Vetterling W.T., Teukolsky S.A., and Flannery B.P. (1992). *Numerical Recipes in C: The art of scientific computing.* Cambridge University Press, Cambridge, UK, 2 edition. 19

Rapberger R., Lukas A., and Mayer B. (2007). Identification of discontinuous antigenic determinants on proteins based on shape complementarities. *J Mol Recognit*, 20(2):113–21. 92

Richardson J.S. (1981). The anatomy and taxonomy of protein structure. *Advances in Protein Chemistry*, 34:167–339. 45, 46, 56

Rögnvaldsson T. and You L. (2004). Why neural networks should not be used for HIV-1 protease cleavage site prediction. *Bioinformatics*, 20(11):1702–9. 81

Rose G.D.R., Gierasch L.M.G., and Smidt J.A.S. (1985). Turns in peptides and proteins. *Advances in Protein Chemistry*, 37:1–109. 48, 56, 57

Rost B. (1996). PHD: predicting one-dimensional protein structure by profile-based neural networks. *Methods Enzymol*, 266:525–39. 29, 37

Rost B. and Sander C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*, 232:584–584. 38

Rumelhart D., Hinton G., and Williams R. (1986). *Learning internal representations by error propagation*, pages 318–363. MIT Press Cambridge. 9, 39, 69

Schechter I. and Berger A. (1967). On the size of the active site in proteases. I. Papain. *Biochemical and Biophysical Research Communications*, 27(2):157 – 162. 81

Schneider T.D. and Stephens R.M. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic Acids Research*, 18(20):6097–6100. 16

Shannon C. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27:379–423, 623–656. 16

Shen H.B.B. and Chou K.C.C. (2008). HIVcleave: a web-server for predicting human immunodeficiency virus protease cleavage sites in proteins. *Anal Biochem*, 375(2):388–90. 82

Shepherd A.J., Gorse D., and Thornton J.M. (1999). Prediction of the location and type of β-turns in proteins using neural networks. *Protein Science*, 8(5):1045–1055. 57, 61

Song J., Tan H., Takemoto K., and Akutsu T. (2008). HSEpred: predict half-sphere exposure from protein sequences. *Bioinformatics*, 24(13):1489–97. 92

Spearman C. (1904). The proof and measurement of association between two things. By C. Spearman, 1904. *The American journal of psychology*, 100(3-4):441. 41

Sweredoski M.J. and Baldi P. (2008). PEPITO: improved discontinuous B-cell epitope prediction using multiple distance thresholds and half sphere exposure. *Bioinformatics*, 24(12):1459–60. 92, 95, 96, 101

Swets J.A. (1996). *Signal detection theory and ROC analysis in psychology and diagnostics.* Lawrence Erlbaum Associates Mahwah, NJ. 70

Thompson J.D., Gibson T.J., Plewniak F., Jeanmougin F., and Higgins D.G. (1997). The CLUSTAL_X windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic Acids Res*, 25(24):4876–82. 83

Thompson T.B., Chou K.C., and Zheng C. (1995). Neural network prediction of the HIV-1 protease cleavage sites. *J Theor Biol*, 177(4):369–79. 81

Van Regenmortel M.H. and Pellequer J.L. (1994). Predicting antigenic determinants in proteins: looking for unidimensional solutions to a three-dimensional problem? *Pept Res*, 7(4):224–8. 91

Venkatachalam C.M. (1968). Stereochemical criteria for polypeptides and proteins. V. Conformation of a system of three linked peptide units. *Biopolymers*, 6(10):1425–1436. 45, 46, 56

Wang G. and Dunbrack Jr R.L. (2003). PISCES: a protein sequence culling server. *Bioinformatics*, 19(12):1589. 38, 41, 66

Wang J.Y., Lee H.M., and Ahmad S. (2007). SVM-Cabins: prediction of solvent accessibility using accumulation cutoff set and support vector machine. *Proteins: Structure, Function, and Bioinformatics*, 68(1):82–91. 30

Wensing A.M.J., van Maarseveen N.M., and Nijhuis M. (2010). Fifteen years of HIV Protease Inhibitors: raising the barrier to resistance. *Antiviral Res*, 85(1):59–74. 80

Xu Z., Zhang C., Liu S., and Zhou Y. (2006). QBES: predicting real values of solvent accessibility from sequences by efficient, constrained energy optimization. *Proteins: Structure, Function, and Bioinformatics*, 63(4):961–966. 30

Yuan Z., Burrage K., and Mattick J.S. (2002). Prediction of protein solvent accessibility using support vector machines. *Proteins: Structure, Function, and Bioinformatics*, 48(3):566–570. 30

Yuan Z. and Huang B. (2004). Prediction of protein accessible surface areas by support vector regression. *Proteins: Structure, Function, and Bioinformatics*, 57(3):558–564. 30, 32, 33

Zhang C.T. and Chou K.C. (1997). Prediction of beta-turns in proteins by 1-4 and 2-3 correlation model. *Biopolymers*, 41(6):673–702. 56, 61

Zhang Q., Yoon S., and Welsh W.J. (2005). Improved method for predicting beta-turn using support vector machine. *Bioinformatics*, 21(10):2370. 57

Zheng C. and Kurgan L. (2008). Prediction of beta-turns at over 80% accuracy based on an ensemble of predicted secondary structures and multiple alignments. *BMC bioinformatics*, 9(1):430. 57, 61

Appendices

Supplementary table S1: Maximum Accessible Surface Area (ASA$_{max}$)

Amino acid			Gly-x-Gly	Ala-x-Ala
Alanine	Ala	A	115	110.2
Arginine	Arg	R	225	229.0
Aspartic Acid	Asp	D	150	144.1
Asparagine	Asn	N	160	146.4
Cysteine	Cys	C	135	140.4
Glutamic Acid	Glu	E	190	174.7
Glutamine	Gln	Q	180	178.6
Glycine	Gly	G	75	78.7
Histidine	His	H	195	181.9
Isoleucine	Ile	I	175	185.0
Leucine	Leu	L	170	183.1
Lysine	Lys	K	200	205.7
Methionine	Met	M	185	200.1
Phenylalanine	Phe	F	210	210.7
Proline	Pro	P	145	141.9
Serine	Ser	S	115	117.2
Threonine	Thr	T	140	138.7
Tryptophan	Trp	W	255	240.5
Tyrosine	Tyr	Y	230	213.7
Valine	Val	V	155	153.7

Suppplementary Table S1 lists the maximum possible accessible surface area measured in Å^2, for the given amino acid located in the center of a tri-peptide flanked by either glycine (column 4, Chothia (1976)) or alanine (column 5, Ahmad et al. (2003)).

Supplementary table S2: Papers citing NetSurfP

#	Reference
1	Afzal, S., Idrees, M., Ali, M., Ilyas, M., Hussain, A., Akram, M., . . . Shahid, M. (2011). Envelope 2 protein phosphorylation sites S75 & 277 of hepatitis C virus genotype 1a and interferon resistance: A sequence alignment approach. Virology Journal, 8(1), 71. doi:10.1186/1743-422X-8-71
2	Baek, J. H., Ji, Y., Shin, J. S., Lee, S., & Lee, S. H. (2010). Venom peptides from solitary hunting wasps induce feeding disorder in lepidopteran larvae. Peptides. doi:10.1016/j.peptides.2010.12.007
3	Blicher, T., Gupta, R., Wesolowska, A., Jensen, L. J., & Brunak, S. (2010). Protein annotation in the era of personal genomics. Current Opinion in Structural Biology, 20(3), 335-41. doi:10.1016/j.sbi.2010.03.008
4	Bulgheresi, S., Gruber-Vodicka, H. R., Heindl, N. R., Dirks, U., Kostadinova, M., Breiteneder, H., & Ott, J. A. (2011). Sequence variability of the pattern recognition receptor mermaid mediates specificity of marine nematode symbioses. The ISME Journal. doi:10.1038/ismej.2010.198
5	Castori, M., Castiglia, D., Brancati, F., Foglio, M., Heath, S., Floriddia, G., . . . Zambruno, G. (2011). Two families confirm schöpf-schulz-passarge syndrome as a discrete entity within the WNT10A phenotypic spectrum. Clinical Genetics, 79(1), 92-95. doi:10.1111/j.1399-0004.2010.01513.x
6	Chen, W. J., Guo, C. J., Zhou, Z. C., Yuan, L. Q., Xiang, Z. M., Weng, S. P., . . . He, J. G. (2011). Molecular cloning of ikk from the mandarin fish siniperca chuatsi and its up-regulation in cells by ISKNV infection. Veterinary Immunology and Immunopathology, 139(1), 61-6. doi:10.1016/j.vetimm.2010.07.025
7	Chen, Y., Ku, W., Lin, P., Chou, H., Khoo, K., & Chen, Y. (2010). An s-alkylating labeling strategy for site-specific identification of the s-nitrosoproteome. Journal of Proteome Research, 0(ja). doi:10.1021/pr100680a
8	Danielsen, J. M., Sylvestersen, K. B., Bekker-Jensen, S., Szklarczyk, D., Poulsen, J. W., Horn, H., . . . Nielsen, M. L. (2010). Mass spectrometric analysis of lysine ubiquitylation reveals promiscuity at site level. Molecular & Cellular Proteomics : MCP. doi:10.1074/mcp.M110.003590
9	Dubois, V., Lambeir, A. M., Vandamme, S., Matheeussen, V., Guisez, Y., Scharpé, S., De Meester, I. (2010). Dipeptidyl peptidase 9 (DPP9) from bovine testes: Identification and characterization as the short form by mass spectrometry. Biochimica Et Biophysica Acta, 1804(4), 781-8. doi:10.1016/j.bbapap.2009.11.022
10	Genisyuerek, S., Papatheodorou, P., Guttenberg, G., Schubert, R., Benz, R., & Aktories, K. (2011). Structural determinants for membrane insertion, pore formation and translocation of clostridium difficile toxin B. Molecular Microbiology, no-no. doi:10.1111/j.1365-2958.2011.07549.x
11	Hamdi-Cherif, A. (2010). Integrating machine learning in intelligent bioinformatics. [Integrating Machine Learning in Intelligent Bioinformatics] Wseas Transactions on Computers, 9(4).
12	Herrmann, H. H., & Strelkov, S. V. S. (2011). History and phylogeny of intermediate filaments: Now in insects. [History and phylogeny of intermediate filaments: Now in insects] BMC Biology, 9(16). doi:doi:10.1186/1741-7007-9-16
13	Ingale, A. (2010). Antigenic epitopes prediction and MHC binder of a paralytic insecticidal toxin (ITX-1) of tegenaria agrestis (hobo spider). Open Access Bioinformatics.
14	Jørgensen, K. W., Buus, S., & Nielsen, M. (2010). Structural properties of MHC class II ligands, implications for the prediction of MHC class II epitopes. Plos One, 5(12), e15877.
15	Kazmier, K., Alexander, N., Meiler, J., & Mchaourab, H. (2010). Algorithm for selection of optimized EPR distance restraints for de novo protein structure determination. Journal of Structural Biology. doi:10.1016/j.jsb.2010.11.003

16	Kim, J. K., Lee, J. R., Kang, J. W., Lee, S. J., Shin, G. C., Yeo, W. S., . . . Kim, K. P. (2010). Selective enrichment and mass spectrometric identification of nitrated peptides using fluorinated carbon tags. Anal Chem. doi:10.1021/ac102080d
17	Lambertsen, L., & Kerrn, M. (2010). Test of a novel streptococcus pneumoniae serotype 6 C type specific polyclonal antiserum(factor antiserum 6 d) and characterisation of serotype 6 C isolates in denmark. BMC Infectious Diseases, 10(1), 282.
18	Laure, L., Danièle, N., Suel, L., Marchand, S., Aubert, S., Bourg, N., . . . Richard, I. (2010). A new pathway encompassing calpain 3 and its newly identified substrate cardiac ankyrin repeat protein is involved in the regulation of the nuclear factor- b pathway in skeletal muscle. FEBS Journal, 4322-4337. doi:10.1111/j.1742-4658.2010.07820.x
19	Lee, S. A., Belyaeva, O. V., & Kedishvili, N. Y. (2011). Evidence that proteosome inhibitors and chemical chaperones can rescue the activity of retinol dehydrogenase 12 mutant T49M. Chemico-Biological Interactions. doi:10.1016/j.cbi.2011.01.001
20	Li, L., Orner, B. P., Huang, T., Hinck, A. P., & Kiessling, L. L. (2010). Peptide ligands that use a novel binding site to target both tgf- receptors. Molecular Biosystems, 6(12), 2392-402. doi:10.1039/c0mb00115e
21	Lozano-Juste, J., Colom-Moreno, R., & León, J. (2011). In vivo protein tyrosine nitration in arabidopsis thaliana. Journal of Experimental Botany. doi:10.1093/jxb/err042
22	McGinnes, C. T. M. (2010). Characterization and evolution of the SERH immobilization antigen genes in tetrahymena thermophila. Thesis, Cleveland State University.
23	Meier, I., Zhou, X., Brkljacić, J., Rose, A., Zhao, Q., & Xu, X. (2010). Targeting proteins to the plant nuclear envelope. Biochemical Society Transactions, 38(3), 733. doi:10.1042/BST0380733
24	Meshkin, A., & Ghafuri, H. (2010). Prediction of relative solvent accessibility by support vector regression and best-first method. Excli Journal, 9, 29-38.
25	Möbius, K., Arias-Cartin, R., Breckau, D., Hännig, A., Riedmann, K., Biedendieck, R., . . . Jahn, D. (2010). Heme biosynthesis is coupled to electron transport chains for energy generation. Proceedings of the National Academy of Sciences, 107(23), 10436 -10441. doi:10.1073/pnas.1000956107
26	Murray, C. I., Kane, L. A., Uhrigshardt, H., Wang, S. B., & Van Eyk, J. E. (2010). Site-Mapping of in vitro s-nitrosation in the cardiac mitochondrial: Implications for cardioprotection. Molecular & Cellular Proteomics : MCP. doi:10.1074/mcp.M110.004721
27	Neema, M., Karunasagar, I., & Karunasagar, I. (2011). Antigenic epitopes and MHC binders in OMP A of fish pathogen edwardsiella tarda: A bioinformatic study. Biotechnology, Bioinformatics, Bioengineering. doi:10.1186/1745-7580-2-2
28	Nielsen, M., Lundegaard, C., Lund, O., & Petersen, T. (2010). Cphmodels-3.0 - remote homology modeling using structure-guided sequence profiles. Nucl. Acids Res., 38(suppl_2), W576-581. doi:10.1093/nar/gkq535
29	Noorbakhsh, R., Mortazavi, S. A., Sankian, M., Shahidi, F., Assarehzadegan, M. A., & Varasteh, A. (2010). Cloning, expression, characterization, and computational approach for cross-reactivity prediction of manganese superoxide dismutase allergen from pistachio nut. Allergol International, 59(3), 295-304. doi:10.2332/allergolint.10-OA-0174
30	Nunes, V. S., Damasceno, J. D., Freire, R., & Tosi, L. R. (2011). The hus1 homologue of leishmania major encodes a nuclear protein that participates in DNA damage response. Molecular and Biochemical Parasitology, 177(1), 65-9. doi:10.1016/j.molbiopara.2011.01.011
31	Pedace, L., Castiglia, D., De Simone, P., Castori, M., De Luca, N., Amantea, A., . . . De Bernardo, C. (2011). AXIN2 germline mutations are rare in familial melanoma. Genes, Chromosomes & Cancer. doi:10.1002/gcc.20855
32	Petersen, B., Lundegaard, C., & Petersen, T. N. (2010). Netturnp - neural network prediction of beta-turns by use of evolutionary information and predicted protein sequence features. Plos One, 5(11), e15079. doi:10.1371/journal.pone.0015079
33	Rahbar, M. R., Rasooli, I., Mousavi Gargari, S. L., Amani, J., & Fattahian, Y. (2010). In silico analysis of antibody triggering biofilm associated protein in acinetobacter baumannii. Journal of Theoretical Biology, 266(2), 275-90. doi:10.1016/j.jtbi.2010.06.014

34	Rodríguez, M., Sánchez, O., & Alméciga-Díaz, C. (2010). Gene cloning and enzyme structure modeling of the aspergillus oryzae N74 fructosyltransferase. Molecular Biology Reports. doi:10.1007/s11033-010-0213-0
35	Samson, A. O., & Levitt, M. (2011). Normal modes of prion proteins: From native to infectious particle. Biochemistry. doi:10.1021/bi1010514
36	Sharpe, L. J., Luu, W., & Brown, A. J. (2011). Akt phosphorylates sec24: New clues into the regulation of er-to-golgi trafficking. Traffic (Copenhagen, Denmark), 12(1), 19-27. doi:10.1111/j.1600-0854.2010.01133.x
37	Sun, M., Hillmann, P., Hofmann, B., Hart, J., & Vogt, P. (2010). Cancer-Derived mutations in the regulatory subunit p85 of phosphoinositide 3-kinase function through the catalytic subunit p110. Proceedings of the National Academy of Sciences of the United States of America. doi:10.1073/pnas.1009652107
38	Via, A., Gould, C., Gemund, C., Gibson, T., & Helmer-Citterich, M. (2009). A structure filter for the eukaryotic linear motif resource. BMC Bioinformatics, 10(1), 351. doi:10.1186/1471-2105-10-351
39	Wang, S., Wu, Y., & Outten, F. W. (2011). Fur and the novel regulator yqji control transcription of the ferric reductase gene yqjh in escherichia coli. Journal of Bacteriology, 193(2), 563-74. doi:10.1128/JB.01062-10
40	Xu, L., Li, Y., Haworth, I., & Davies, D. (2010). Functional role of the intracellular loop linking transmembrane domains 6 and 7 of the human dipeptide transporter hpept1. The Journal of Membrane Biology. doi:10.1007/s00232-010-9317-7
41	Zhao, J., Sun, Z., Yang, H., Zhang, C., Yu, X., Wen, Z., . . . Zhang, S. (2010). Cloning, expression and immunological evaluation of a short fragment from rv3391 of mycobacterium tuberculosis. Annals of Microbiology. doi:10.1007/s13213-010-0148-7

Supplementary table S3. Setups tested for training in the second layer networks

Setup	pssm	sec-rsa	*β*-turn-G	*β*-turn-P
A	×	×		
B	×	×	×	
C	×		×	
D	×			×
E				×
F		×	×	
G			×	
H	×	×		×
I		×		×
J	×	×	×	×
K	×		×	×
L			×	×
M		×	×	×

Setups tested for training in the second layer networks. The table is listing the different setups tested for training in the second layer networks. In the table abbreviations are as follows: β-turn-G = β-turn/not-β-turn prediction from first layer networks, β-turn-P = position specific predictions from first layer networks, sec-rsa = secondary structure and surface accessibility predictions from NetSurfP (Petersen et al. (2009)), PSSM = Position Specific Scoring Matrices.

Supplementary table S4. Test performance for the first layer β-turn-P networks

β-turn-P networks	Q_{total}	PPV	Sens	Spec	MCC	AUC
β-turn-P (Position 1)	84.0	26.1	65.9	85.4	0.34	0.849
β-turn-P (Position 2)	83.5	25.7	67.4	84.8	0.34	0.852
β-turn-P (Position 3)	83.5	25.7	67.2	84.8	0.34	0.852
β-turn-P (Position 4)	83.5	25.6	67.0	84.8	0.34	0.851

Test performance for the first layer β-turn-P networks. Test performances from the first layer β-turn-P networks using the Cull-2220 dataset. All performance measures have been explained in the methods section. All β-turn-P networks were trained using pssm + sec + rsa, where pssm = Position Specific Scoring Matrix, sec = Secondary structure predictions (Petersen et al. (2009)), rsa = Relative solvent accessibility predictions (Petersen et al. (2009)). The positions in the four network trainings are referring to the position in a β-turn.

Supplementary table S5. Test performances from the first and second layer β-turn-G networks

β-turn-P networks	**$\mathbf{Q_{total}}$**	**PPV**	**Sens**	**Spec**	**MCC**	**AUC**
First layer networks	77.8	51.3	73.1	79.1	0.47	0.846
Second layer networks	78.8	53.0	71.5	81.0	0.48	0.849

Test performances from the first and second layer β-turn-G networks using the Cull-2220 dataset. All performance measures have been explained in the methods section. The first layer networks were using pssm + sec + rsa, and the secondary networks were using β-turn-P + β-turn-G + sec + rsa, where the used nomenclature are: pssm = Position Specific Scoring Matrix, sec = secondary structure predictions (Petersen et al. (2009)), rsa = relative solvent accessibility predictions (Petersen et al. (2009)). β-turn-G = β-turn/non-β-turn predictions, β-turn-P = predictions from the position specific networks.

Supplementary table S6. Amino acid statistics in Cull-2200 dataset

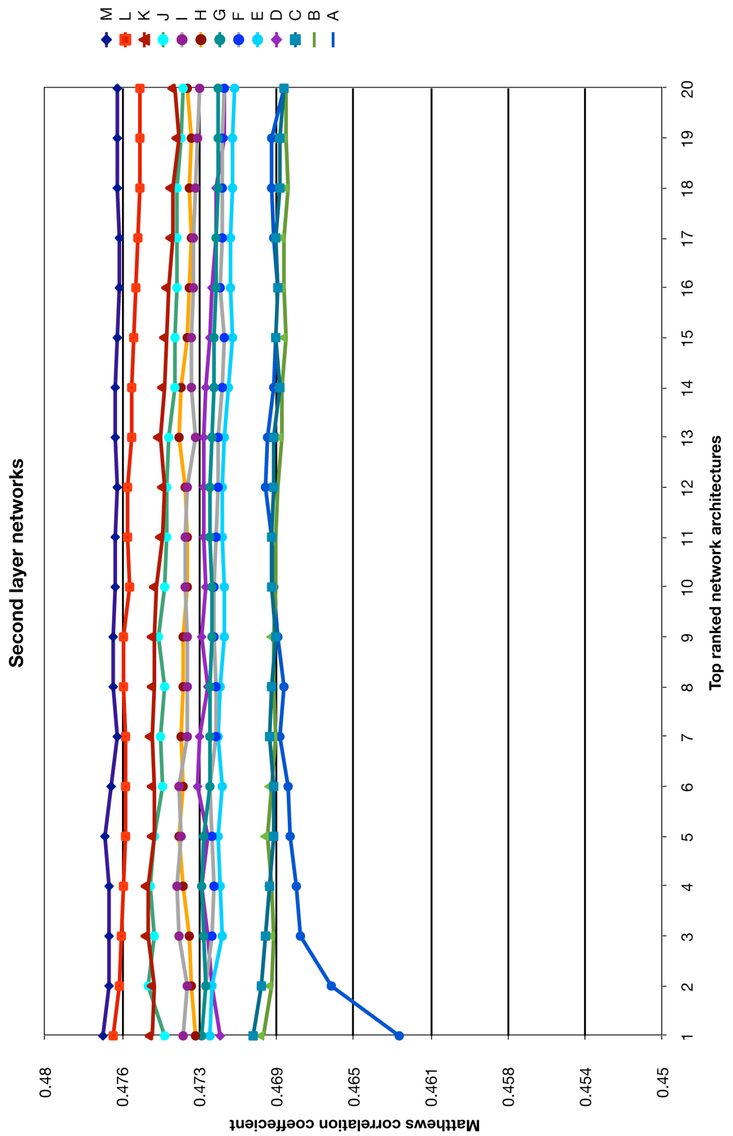
		β**-turn statistics**		**Amino acid statistics**	
Amino Acid		**Amount**	**% of all**	**Amount**	**% of all**
Ala	A	6,277	6.4	35,923	7.95
Cys	C	1,447	1.5	6,081	1.35
Asp	D	8,779	8.9	26,660	5.90
Glu	E	6,240	6.3	31,514	6.98
Phe	F	3,378	3.4	18,155	4.02
Gly	G	11,426	11.6	32,443	7.18
His	H	2,627	2.7	12,206	2.70
Ile	I	3,416	3.5	25,623	5.67
Lys	K	5,636	5.7	26,234	5.81
Leu	L	5,853	5.9	41,056	9.09
Met	M	1,107	1.1	10,605	2.35
Asn	N	6,254	6.3	19,059	4.22
Pro	P	6,865	7.0	20,693	4.58
Gln	Q	3,364	3.4	17,489	3.87
Arg	R	4,505	4.6	23,249	5.15
Ser	S	6,998	7.1	27,414	6.07
Thr	T	5,417	5.5	24,113	5.34
Val	V	4,811	4.9	31,213	6.91
Trp	W	1,224	1.2	6,353	1.41
Tyr	Y	3,018	3.1	15,723	3.48
Total		**98,642**	**100**	**451,806**	**100**

Amino acid statistics in Cull-2200 dataset. Frequencies for amino acids in β-turns and the Cull-2220 training set. The first part of the table 'β-turn statistics' shows the amount of residues, which have been assigned as β-turns and their percentage of the total amount of β-turn assigned residues in the Cull-2220 set. The second part of the table 'Amino acid statistics' shows the amount of residues and the percentage of the total Cull-2220 set.

Supplementary table S7. Dihedral angles for the β-turn types as used by PROMOTIF

β-turn types	**Φ, Ψ *(i + 1)***	**Φ, Ψ *(i + 2)***
I	-60°, -30°	-90°, 0°
I’	60°, 30°	90°, 0°
II	-60°, 120°	80°, 0°
II’	60°, -120°	-80°, 0°
VIII	-60°, -30°	-120°, 120°
VIa1	-60°, 120°	-90°, 0°
VIa2	-120°, 120°	-60°, 0°
VIba	-135°, 135°	-75°, 160°
IV	β-turns excluded from the above categories	

Dihedral angles for the β-turn types as used by PROMOTIF. Dihedral angles for the β-turn types between residues two (*i+1*) and three (*i+2*) as used by PROMOTIF (Hutchinson and Thornton (1996)). These angles are allowed to deviate by ± 30° from the defined angles, with the addition that one dihedral angle is allowed to deviate as much as ± 40°. Type IV is used for all β-turns, which do not fall within the dihedral angle ranges for the eight defined types. Type VIa1, VIa2 also require a cis-proline at position *i+2*.



Supplementary figure S1. Matthews correlation using different setups and an increasing number of trained network architectures. The figure shows test performances in Matthewss correlation coefficient when including an increasing number of trained networks architectures, named Top ranked network architectures, based on test set performance using different setups. Abbreviations for the setups are as follows: β-turn-P = position specific first layer predictions, β-turn-G = general β-turn/not-β-turn first layer predictions, sec-rsa = secondary structure and surface accessibility predictions from NetSurfP (Petersen et al. (2009)), PSSM = Position Specific Scoring Matrices. The setups are composed as follows: A = PSSM + sec-rsa, B =PSSM + β-turn-G+ sec-rsa, C=PSSM + β-turn-G, D= PSSM + β-turn-P, E= β-turn-P, F = β-turn-G + sec-rsa, G= β-turn-G, H= PSSM + β-turn-P + sec-rsa,I = β-turn-P + sec-rsa, J = PSSM + β-turn-P + β-turn-G + secrsa, K= PSSM + β-turn-P + β-turn-G, L = β-turn-P + β-turn-G, M= β-turn-P + β-turn-G + sec-rsa.

NetSurfP Manual

Included below is the NetSurfP manual as of 15 October 2010, the version which is shipped along with the NetSurfP 1.0 package.

```
NAME
       netsurfp - predict surface accessibility and secondary structure of protein residues

SYNOPSIS
       netsurfp [-a] [-c #] [-d database] [-h] [-i infile]
                [-j sspfile] [-k] [-o outfile] [-s syn_dir]
                [-t FASTA|HOW|PROF] [-v] [-J sspoutfile]

DESCRIPTION
       netsurfp  predicts the surface accessibility and secondary structure of residues in amino
       acid sequences. For each residue the relative exposure is predicted alongside the secondary
     structure; the reliability of the surface accessibility prediction is stated in the form of a Z-score.

       The method has been trained on more than 410,000 amino acid residues, where 44.2% were
       classified as exposed. It was evaluated on an independent dataset.  The performance obtained
       was 79.0% correct prediction with a MCC of 0.577 and PCC of 0.70. The method is described in
       detail in the reference quoted below.

     The input for the prediction is taken from infile (-i) or, if no file is specified, from stdin. The output
       will go to stdout by default; it can also be printed to a file (-o).

       The netsurfp prediction process in in three steps:

       1. Generation of alignment profiles: the input sequences are
          aligned to a non-redundant BLAST database. This step is
          not performed if the input contains profiles already (see
          Input below).

       2. Prediction of the secondary structure. This step in not
          performed if the secondary structure of the input
          sequences is provided by the user (-j).

       3. Prediction of the surface accessibility. This step is
          always performed.

       Input

       The following input formats are supported (-t):

           FASTA (default)
           HOW
           PROF (Position Specific Scoring Matrix)

       The FASTA and HOW formats are described on the manual pages for those programs, respectively.

     If the alignment profiles of the input sequences are already available the PROF format should be used
       instead; it consists of lines in the form:

             1: Assignment B (buried) or E (exposed)
             2: Amino acid residue in one-letter code
             3: Sequence name
             4: Residue number
          5-24: Log-odds for the probabilities of the 20 amino
                acids on that position

       A PROF file can be generated by blastpgp (see that software).

     If a position specific scoring matrix is not provided by the user, it will be generated by the netsurfp
       software in the course of the prediction process (see above,step 2).

      The input sequences have to have unique identifiers. Special characters in the identifiers will be
```

```
       replaced by the underscore characters ("_").



       Output

       The output is in columns, as follows:

          1: Class assignment - B (buried) or E (exposed),
             see below
          2: Amino acid
          3: Sequence name
          4: Amino acid number
          5: Relative Surface Accessibility - RSA
          6: Absolute Surface Accessibility
          7: Z-fit score (RSA prediction only)
          8: Probability for alpha-helix
          9: Probability for beta-strand
         10: Probability for coil

      The buried/exposed assignment has the threshold of 25% exposure, based on the first layer networks
       and not on RSA.

      The secondary structure predictions (columns 8-10) are only included if required specifically (-a).

       The output can be made more verbose (-v). All the reporting goes to stderr.

OPTIONS
       -a     include the secondary structure prediction in the output. The default is not to do this.

      -c #   process # sequences in parallel. The default is 2. This option needs to be used with caution as
       processing one sequence may employ several cpu:s depending of the local setup.

       -d     path to local nr (non-redundant) blast database. For speed purposes nr70 can be used.

       -h     Show the allowed command line syntax and exit.

       -i infile
         The input file to process. If not specified, stdin will be used instead. The format must be FASTA
              (default), HOW or PROF.

       -j sspfile
         use the secondary structure predictions in sspfile rather than the native netsurfp predictions.

       -k     keep all temporary directories. The default is not to do this.

       -o outfile
              Write the output to outfile. If not specified stdout will be used.

       -s syndir
         The default location of the synapse file. Do not change this unless you know what you are doing.

       -t format
         input file format. If the default FASTAis not used, this option must be given as either HOW or PROF
(see above).

       -v     Verbose mode. All reporting will go to stderr.

       -J  sspoutfile
         Generate an extra output file with the sequences and the secondary structure predictions. The default
              is not to do this.

EXAMPLES
        netsurfp -h
              this will show the help menu.

        netsurfp -i ./test/test.fasta
              a fasta file will be used as input, and rsa results will be written to stdout.
        netsurfp -i ./test/test.fasta -a
          a fasta file will be used as input, and rsa results including secondary structure predictions will
               be written to stdout.

       netsurfp -i ./test/test.prof -t PROF -a -v -k -o file.rsa
          a PROF file will be used as input. Secondary structure predictions will be included in the result file,
               file.rsa, temp directories will be kept and  verbose mode is on.


VERSION
       This manpage describes netsurfp 1.0.


REFERENCE
       For publication of results, please cite:

     A generic method for assignment of reliability scores applied to solvent accessibility predictions.
      Bent Petersen, Thomas Nordahl Petersen, Pernille Andersen, Morten Nielsen and Claus Lundegaard.
       BMC Structural Biology, 9:51, 2009

       The method is also available on-line at:

       http://www.cbs.dtu.dk/services/NetSurfP/


AUTHOR
       Bent Petersen, bent@cbs.dtu.dk, April 2010.
       Last updated in April 2010 (v. 1.0).

FILES
       /usr/cbs/bio/bin/netsurfp           executable
       /usr/cbs/bio/src/netsurfp-1.0    software home

SEE ALSO
       fasta(1), how(1)
```