Protein Fold Class Prediction:

New Methods of Statistical Classification

Janet Grassmann[1)], Martin Reczko[2)], Sandor Suhai[3)] and Lutz Edler[4)]

[1)]Department of Statistics, Stanford University, Palo Alto, CA 94305-4065, U.S.A.
[2)]Synaptic Ltd. Aristotelous 313, 13671 Acharnai, Greece
[3)]Department of Molecular Biophysics, German Cancer Research Center
[4)]Biostatistics Unit R0700, German Cancer Research Center,
D-69120 Heidelberg, Germany
e-mail: edler@dkfz-heidelberg.de

Abstract

Feed forward neural networks are compared with standard and new statistical classification procedures for the classification of proteins. We applied logistic regression, an additive model and projection pursuit regression from the methods based on a posterior probabilities; linear, quadratic and a flexible discriminant analysis from the methods based on class conditional probabilities, and the K-nearest-neighbors classification rule. Both, the apparent error rate obtained with the training sample (n=143) and the test error rate obtained with the test sample (n=125) and the 10-fold cross validation error were calculated. We conclude that some of the standard statistical methods are potent competitors to the more flexible tools of machine learning.

Keywords: protein fold class, statistical classification, prediction accuracy, neural networks, discrimination, regression, dipeptide matrix, confusion matrix

Introduction

Knowledge of the 3D structure of a protein is essential for describing and understanding its function and its use in molecular modeling. In order to bridge the gap between the large number of sequenced proteins and the much smaller number of proteins whose structure has been determined, artificial neural networks have been intensively used for predicting secondary and tertiary structural aspects from amino acid sequence data. For a recent summary for secondary structure prediction and for tertiary structure prediction we refer to Finkelstein (1997) and Rost and O'Donoghue (1997). In this study, we investigated classification methods, including the feedforward neural networks, for the best discrimination and prediction of protein fold classes. From a data base of protein sequence data of Reczko and Bohr (1994) we used a set of 268 sequences and their classification into 42 fold classes. The methods described next are compared in Section 3. Correctly and wrongly predicted sequences are exhibited in the form of confusion matrices. The results are discussed in Section 4.

Classification Problem and Methods

Proteins and their Classes

A protein of length N is formally represented as the sequence

$$P=(s_1, \ldots, s_1),\ s_i \in \{A_1, \ldots, A_{20}\},$$

when A_j, j=1,...,20, denote the twenty different amino acids. Proteins can be classified by secondary structure elements. A sequence $P = (s_1, \ldots, s_N)$, is associated with a secondary sequence $Q = (r_1, \ldots, r_N)$ where each r_i belongs to one of the three secondary classes α-helix, β-sheet or coil. Next towards tertiary, structure is a four class classification of proteins based on the presence or absence of α-helices and the β-sheets in the secondary sequence. One distinguishes four classes {only α}, {only β}, {one part α plus one part β ($\alpha+\beta$)}, and {α and β alternating (α/β)} (Lesk 1991). Subsequently, this will be called the super-secondary classification SSC. Considering the topological similarity of the backbone, a fold class definition proposed by Pascarella and Argos (1992) of 38 fold classes has been enlarged up to 42 classes of tertiary structure by Reczko and Bohr (1994), subsequently denoted 42-CAT. In the following we will refer to these classes exclusively by their numbers ranging from 1 to 42.

Data

In this analysis we used the protein sequences of Reczko and Bohr (1994), listed in http://www.dkfz-heidelberg.de/biostatistics/protein/gsme97.html. This data set was subdivided randomly into a training set of 143 and a test set of 125 sequences balanced with respect to the prior probabilities of the 42 classes. Each test sequence was compared with the training set sequences for sequence similarity and sequence identity. Homology ranged between less than 10% (9 cases) and 100% (12 cases). Within the training set and within the test set pairwise sequence identity was less than 50%, respectively. For a listing of the training and the test set and sequence similarity see the www-document above.

Statistical Classification

Structural classification of a protein consists in finding a classification rule **R** mapping each amino acid sequence P from a sequenced space Ξ into the finite set of structural classes **Y:**

$$\mathbf{R}: \quad \Xi \quad \rightarrow \quad \mathbf{Y} = \{1, ..., k\},$$

where K denotes the number of classes. Ξ can be considered as the set of presently available protein sequences as e.g. the SWISSPROT database and **Y** could be either the 42-CAT or SSCs. Sequence information of the protein is transformed into an input vector x from a more regular feature space **X,** e.g. the amino acid distribution of relative frequencies of length 20, or the pairs of neighboring amino acids

$$x_j^{(2)} = \{ s_j , s_{j+1} \} , j = 1, ... , N-1,$$

and their frequencies represented as a 20 x 20 dipeptide matrix, (Reczko et al. 1994). To reduce the dimension of the measurement space further, a principal component analysis was performed and the first principal components that explain 90% of the variance of the data were used as a third feature space.
Given the input vector $x = (x_1, ... , x_p) \in \mathbf{X}$, the structural classes $k \in \{1, ..., K\}$, the *a priori* class probabilities p(k), the *a posterior* class probability p(k|x) of class k given the protein sequence information x and the class conditional probability p(x|k) of x given the class k the Bayes rule assigns sequence x to that class for which the posterior class probability is maximum.

(1) $\quad d(x) = \{ k \in \{1, ... , K\}: p(k|x) = \max_j p(j|x) \}$

This is equivalent to

(2) $\quad d(x) = \{k \in \{1, ..., K\}:$

$$p (x|k) p(k) = \max_j \{p(x|j) p (j)\}$$

when using the Bayes formula

(3) $\quad p(y|x)p(x) = p(x|y)p(y).$

Hence, classification problem is solved either by modeling the class conditional probabilities p(x|k) or the *a posterior* probabilities p(k|x). Furthermore, the classification problem can easily be formulated as a regression problem, when the class variable Y is transformed into K „dummy" variables Y_k, k=1,...,K, where

(4) $\quad Y_k = 1 \quad \text{if } Y = k \quad \text{and} \quad Y_k = 0 \quad \text{if } Y \neq k .$

From this results the regression model

(5) $\quad f_k(x) = E[Y_k | x] = P(Y_k = 1|x)$

$$= P(Y=k|x) = p(k|x).$$

For details see Ripley (1994). The following classification procedures were applied in this investigation:

(i) FEED FORWARD NEURAL NETWORKS (FNN):
A feedforward neural network is defined by an input vector, output units and weighted connections from the input units via one single layer of hidden units to the output units (Grassmann and Edler 1996). For learning the net we used the Kullback-Leibler distance (Ripley 1994) which is equivalent to the minimization of the negative log-likelihood function. FNNs were applied sequentially starting with no hidden units NN(0) and then increasing the number of hidden units one by one until there was no improvement of the test error or the CV error, respectively. An FNN without hidden units is equivalent to polychotomous logistic regression with

(6) $$f_k(x) = p(k|x) = \frac{\exp(\eta_k(x))}{\sum_{m=1}^{K} \exp(\eta_m(x))}$$

and the linear predictor $\eta_m = \beta_m^T \cdot x$ (Schumacher et al 1994; Grassmann 1996).

(ii) ADDITIVE MODEL (BRUTO):
The additive model of Hastie and Tibshirani (1990) is more appropriate for nonlinear influences of the input variables. Its given by

(7) $$f_k(x) = p(k|x) = a_k + \sum_{j=1}^{p} \phi_{kj}(x_j)$$

where the ϕ's play the role of smoothing functions suitable for non-linearities. We used a special case of this additive model with the ϕ's being smoothing splines. This type of flexible discriminant analysis is known as the BRUTO method (Hastie et al. 1994).

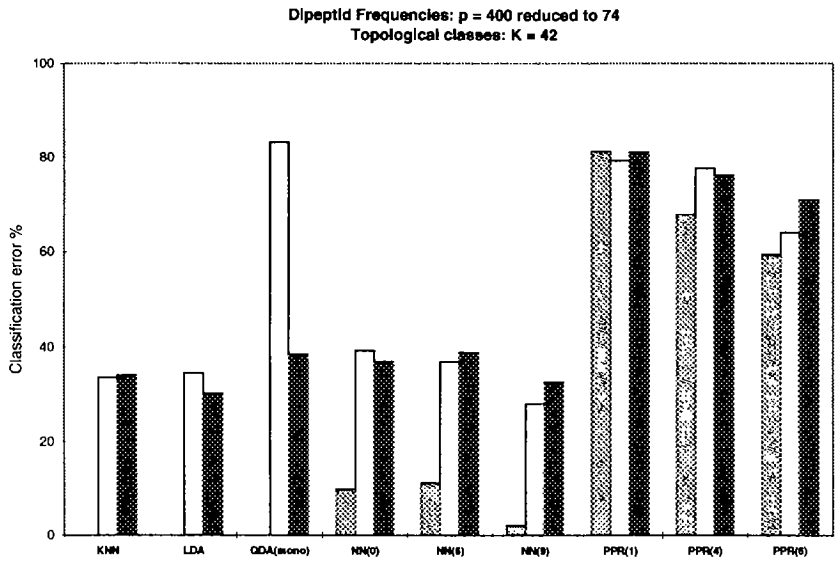

Figure 1:
Comparison of the classification results for the case of the first 74 principle components of the dipeptide distribution as input variables and the 42-class definition 42-CAT. Presented are the misclassification rates for a training data set (hatched), the test data set (clear) and the 10-fold cross-validation error (CV(10)) (dark). The methods are the K-th Nearest Neighbor method (KNN), the linear discriminant analysis (LDA), the quadratic discriminant analysis with interaction terms (QDA), and without interaction terms (QDA-mono), the flexible discriminant analysis using an additive model in the regression part (FDA/BRUTO), the neural networks with n hidden units (NN(n)) and the projection pursuit regression with J additive terms (PPR(J)). A missing bar indicates a zero error rate.

(iii) PROJECTION PURSUIT REGRESSION (PPR):
The projection-pursuit regression of Friedman and Stützle (1981) has the form

(8) $$f_k(x) = p(k_k|x) = \sum_{m=1}^{M} \beta_{km} \phi_{km} (\sum_{i=1}^{p} \alpha_{mi} x_i)$$

where the number of terms M has to be selected appropriately. The ϕ_{km} are chosen adaptively depending on the data.

(iv) LINEAR DISCRIMINANT (LDA):
The linear discriminant analysis assumes that the class conditional probability follows a multivariate Gaussian distribution

(9) $$p(x|j) = 2\pi^{-\frac{p}{2}} \cdot |\Sigma|^{-\frac{1}{2}} \cdot \exp(-\frac{1}{2} \cdot (x - \mu_j)^T \cdot \Sigma^{-1} (x - \mu_j)).$$

A new observation with predictor x_0 is classified to that class with its centroid closest to x_0 with respect to the Mahalanobis distance, using the pooled within-class covariance matrix Σ.

(v) QUADRATIC DISCRIMINANT ANALYSIS QDA):
This is a generalization of the LDA where the covariance matrix Σ depends on the class j_j. We denote by QDA-MONO a QDA without interaction terms. For both LDA and QDA see Ripley (1994).

(vi) FLEXIBLE DISCRIMINANT ANALYSIS (FDA):
This generalizes LDA to a non-parametric post-processing multi-response regression using an optimal scoring technique to improve the classification (Hastie et al 1994).

(vii) K-th-NEAREST-NEIGHBOR CLASSIFICATION RULE (KNN):
Basically, it assigns a new data point x to the class that the majority of the K nearest neighbors of x belong to (Ripley 1994).

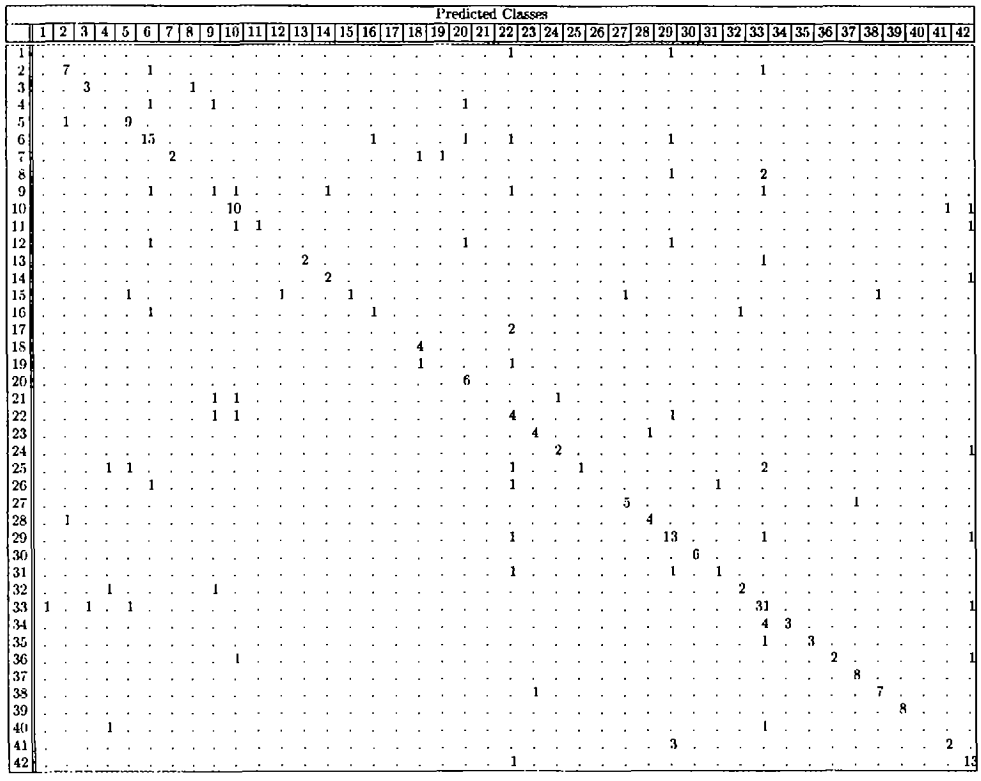

Figure 2:
Confusion table for the cross validation CV(10) of the best FNN with 9 hidden units (NN(9)) for assessing the prediction of the 42 classes of the 268 sequences. The lines represent the true fold classes according to Reczko and Bohr (1994) and the colums represent the predicted fold classes by NN (9). E.g. of the nine proteins of fold class 2 seven are predicted correctly and two are predicted wrongly into class 6 and class 33.

Prediction Error

A naive estimate of the prediction error is the ratio of misclassified sequences of the complete set (apparent prediction error, APE). This error estimate is usually overoptimistic and is reported here for reasons of comparison with earlier published results. Common is a splitting of the data into a training and a test set and the calculation of the ratio of misclassified sequences in the test set only (test error, TPE). More objective is the k-fold cross-validation (CV) error (Efron and Tibshirani 1993) which is less biased preferred for the assessment of the prediction error.

Software

All calculations were performed at a SUN/SPARC station 20 by applying the S-Plus software. The S-libraries *fda*, *nnet* available from the Statlib ftp site were loaded to perform the calculations for the above mentioned regression or classification algorithms. The S-Plus routine *nnet* (Ripley 1994) applies a Newton-Raphson algorithm to fit FNNs. LDA, QDA and FDA/BRUTO were computed with the *fda* function, whereas there exists a function *knn* for running the KNN fitting.

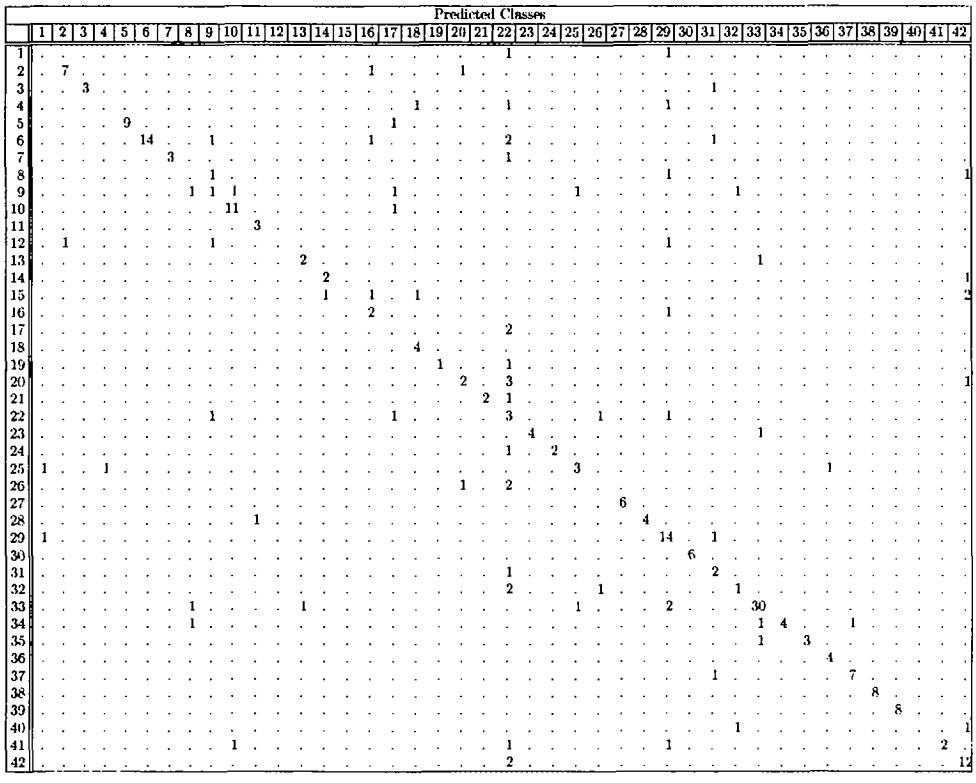

Figure 3:
Confusion table for the cross validation CV(10) of the LDA for the prediction of the 42 classes of the 268 sequences, analogously to Figure 2.

Results

We report here the prediction of the 42 fold classes 42-CAT from the dipeptide frequencies. Results on the classification into the SSCs are given in Edler and Graßmann (1999). Both, for FNN and PPR only a subset of models including the best results is shown by selecting among the numbers of hidden units or the number of additive terms, respectively. When a method was definitely worse than others its results are not reported as it was e.g. the case with the additive model (BRUTO) for 42-CAT.

Since the prediction of the 42-CAT with the 20x20 dipeptide matrix the classification problem would have been over-parametrized with 400 input variables and only 125 cases we reduced the number of input variables by principal component analysis. Figure 1 shows the result of the 42-CAT fold classification based on the first 74 principal components. Classification by linear discriminant analysis (LDA) was best with a 10-fold CV-error rate of 30.2% followed by the best FNN with 9 hidden units and a 10-fold CV-error of 32.5%. The error rate of the KNN was larger (34.0%). A sufficiently large weight decay parameter was needed for the FNN. All other models seemed to be too complex for these data and performed worse. The PPR gave totally disappointing results with error rates larger than 50%. QDA was no more able to provide reasonable prediction results. Interestingly, the NN(9) yielded the lowest test error (28.8%), lower than the KNN (33.6%) and LDA (34.4%).

The performance of the procedures was compared by examining the patterns of correctly and wrongly predicted sequences. For this reason we constructed the so called confusion matrices of dimension 42x42 where the diagonal contains the number of correctly predicted folds and the off-diagonals exhibit the wrongly predicted folds. Here we report the results of the LDA and the NN(9) in the test set. The NN(9) and the LDA diffusion matrices are shown in (Figures 2 and 3). The 5 cytochromes c proteins (class no. 2) e.g. are correctly predicted by the NN(9) in 4 and by the LDA in only 2 cases. The NN(9) predicted one class 2 protein wrongly

into class 6 (globins) which was a more frequent class whereas LDA predicted one class 2 protein wrongly into class 14 (staphylococcal nuclease) and 2 wrongly into class 16 (cytochrome P450). The one protein crambin (class 8) is predicted correctly, both with LDA and NN(9) and all 6 acid proteinases of class 10 are correctly predicted by both methods. Crambin, actually, was to 100% homologue to its partner in the training set, yet the best homology of the 6 acid proteinases to their training partners ranged between 37% and 57% only. The one sulphur protein (class 11) was predicted wrongly by both methods, also the one cytochrome c (class 12). On should notice, however, that the comparison of NN(9) with LDA is impaired by the fact that the result of the NN(9) depends on the starting values. Different starting values may yield slightly different classification errors and may lead to other predictions and therefore to another confusion matrix. The above comparison has to be considered as one of many possible comparisons only.

We investigated also the classification into the four SSCs on the basis of the dipeptide frequencies. The results were qualitatively similar to those for the 42-CAT except that the more flexible nonlinear statistical procedures as PPR gave reasonably low error estimates. However, they were still higher than those obtained with FNNs. NN(10) was best among the FNNs, and slightly superior to the KNN. Compared with the prediction of the SSC on the basis of the 20 amino acid frequencies we got higher misclassification rates for the test set and for the CV, in general. Obviously, we lost information by taking the principal components of the dipeptide matrices instead of the amino acid frequencies.

Discussion

In this study of statistical classification procedures we considered the fold class prediction of proteins on the basis of the amino acid frequencies and especially the dipeptide frequencies. We applied the well-known standard classification procedures KNN, LDA, and QDA and the more flexible methods FDA/BRUTO, PPR and FNN. For the high-dimensional feature space and a large number of classes (K=42) the simplest and most regularized LDA gave the smallest CV error rate, whereas the neural network with 9 hidden units and a weight decay term had the smallest test error, both at about 30%. Neural networks have a number of degrees of freedom which make them on the one hand very flexible, but if too many weights are used the danger of overfitting (fitting the noise) is obvious (Grassmann and Edler 1996). Other difficulties in their application are then sensitivity on starting values, the weight decay parameter, the number of hidden units and the fitting algorithms. Projection pursuit regression (PPR) was disappointing at the first glimpse. Despite its formal similarity with FNNs it could not reach comparable prediction results. An application of biophysical methods (ab initio calculations, potential-energy optimization or threading (see e.g. Neumaier 1997) was beyond the scope of this investigation and further research is needed to compare the statistical methods described above with those.

Our investigation was restricted to a relatively small data set compared with the dimension of the feature space X. The databases which contain 3D information are growing and larger samples sizes become available. But, there still remains the problem of the 'course of dimensionality' because the number of classes is also growing and the class information cannot be described by only a few variables. To reach better performance one has to incorporate even more feature variables. Reasonable choices could be information about the relative site of the respective dipeptide within the sequence or correlations of pairs of amino acids in a more distant neighborhood (distant interactions). A direct inclusion of such information would dramatically increase the number of input variables.

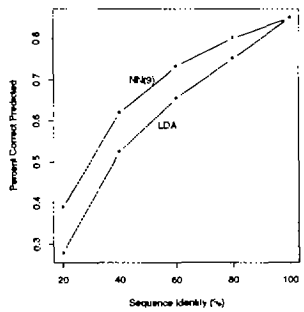

Figure 4:
Percentage of correct prediction of the 42 classes by the FNN with 9 hidden units (NN(9)) and LDA plotted versus sequence identity in the test data set

Finally, there is the question of the choice of sequences for training and for testing. Our full data set included sequences within a big range of homology. When we analyzed the confusion matrices we observed both correct and incorrect classifications of highly homologous sequences as well as for distantly related ones. Figure 3 shows the dependency of the prediction results on the sequence identity. Below 30% sequence identity even the best model could not achieve accuracy of more than 50%. Further research on the influence of the choice of the data and its relation to homology is needed either if the prediction methods above are used or if other methods are applied. Increasing availability of standard data sets will support this research.

We conclude from this study that LDA and KNN can provide reasonably good classification results compared with the much more flexible tools of FNN. There is the need to explore the amino acid sequence for distant

interactions and their inclusion. Also, physico-chemical properties, the occurrence of motifs or evolutionary information (Rost and O'Donoghue 1997) should be taken into account. In practice, the classification methods described in this work can be helpful for screening new sequences and providing hints for their structural shape. As starting point they can reduce the expense and time needed for the estimation of atomic 3D coordinate in original biophysical approaches.

Acknowledgements. For stimulating discussions on statistical classification and prediction and help in applying them the first author is very grateful to Trevor Hastie. A part of this research was supported by the German Academic Exchange Service (DAAD, Doktorandenstipendium HSP II/AUFE).

References

Edler, L.; and Graßmann, J. 1999. Protein prediction is a new field for statistical classification (to appear in Proceedings of the 1997 Joint Summer Research Conference in the Mathematical Sciences on Statistics in Molecular Biology, Seattle,WA).

Efron, B., and Tibshirani, R. J. 1993 *An Introduction to the Bootstrap*. Cambridge: Chapman & Hall.

Finkelstein, A. V. 1997. Protein structure: what is it possible to predict now? *Current Opinions in Structural Biology* 7: 60-71.

Friedman, J. H. 1989. Regularized Discriminant Analysis. *J. Amer. Statist. Assoc.* 84: 165-175.

Friedman, J. H.; and Stützle, W. 1981. Projection pursuit regression. *J. Amer. Statist. Assoc.* 76: 817-823.

Grassmann, J. 1996. Artificial neural networks in regression and discrimination. In: Faulbaum, F. and Bandilla, W. eds: *Softstat '95, Advances in Statistical Software 51*, 399-406. Stuttgart: Lucius & Lucius.

Grassmann, J., and Edler, L. 1996. Statistical classification methods for protein fold class prediction. In: Prat, A. ed. *COMPSTAT. Proceedings in Computational Statistics*, 277-282. Heidelberg: Physica-Verlag.

Hastie, T., and Tibshirani, R. J. 1990. *Generalized Additive Models*. Cambridge: Chapman & Hall.

Hastie, T.; Tibshirani, R. J.; and Buja, A. 1994. Flexible Discriminant Analysis by Optimal Scoring. *J. Amer. Statist. Assoc.* 89: 1255-1270.

Neumaier, A. 1997. Molecular modeling of proteins and mathematical prediction of protein structure. *SIAM Rev* 39: 407-460.

Pascarella, S.; and Argos, P. 1992. A data bank merging related protein structures and sequences. *Protein Engineering* 5: 121-137.

Reczko, M.; and Bohr, H. 1994. The (DEF) Data Base of Sequence Based Protein Fold Class Predictions. *Nucl. Acids Res.* 22: 3616-3619.

Reczko, M.; Bohr, H.; Subramaniam, S.; Pamidighantam, S.; and Hatzigeorgiou, A. 1994. Fold Class Prediction by Neural Networks. In: Bohr, H., and Brunak, S. eds. *Protein Structure by Distance Analysis*. 277-286. Amsterdam: IOS Press.

Ripley, B. D. 1994. Neural networks and related methods for classification. *J. R. Statist. Soc.* B 56: 409-456.

Rost, B.; and O'Donoghue, S. 1997. Sisyphus and prediction of protein structure. *CABIOS* 13: 345-356.

Schumacher, M.; Rossner, R.; and Vach, W. 1994. Neural networks and logistic regression. CSDA Part I and Part II. *Computational Statistics and Data Analysis* 21: 661-682 and 683-701.