# Learning Classifiers for Assigning Protein Sequences to Gene Ontology Functional Families

**Carson Andorf**[1,2,3,5], **Adrian Silvescu**[1,2,3], **Drena Dobbs**[4,5], **Vasant Honavar**[1,2,3,5]
Artificial Intelligence Laboratory[1]
Department of Computer Science[2]
Computational Intelligence, Learning, and Discovery Program[3]
Department of Genetics, Development and Cell Biology[4]
Bioinformatics and Computational Biology Graduate Program[5]
Iowa State University
Ames, IA 50010, USA.
Email: {*andorfc, silvescu, ddobbs, honavar*}*@iastate.edu*

## Abstract

Assigning putative functions to novel proteins and the discovery of sequence correlates of protein function are important challenges in bioinformatics. In this paper, we explore several machine learning approaches to data-driven construction of classifiers for assigning protein sequences to appropriate Gene Ontology (GO) function families using a class conditional probabilistic representation of amino acid sequences. Specifically, we represent protein sequences using class conditional probability distribution of amino acids (amino acid composition) or short ($k$-letter) subsequences ($k$-grams) of amino acids. We compare a model (NB $k$-grams) that ignores the statistical dependencies among overlapping $k$-grams with an alternative, NB($k$), that uses an undirected probabilistic graphical model that captures the relevant dependencies. These two methods require only one pass through the training data during the learning phase, making them especially attractive in settings where there is a need to update the classifiers as new training data become available. We also explore a support vector machine (SVM) classifier, SVM $k$-grams, trained on the $k$-gram class conditional probability distributions of sequences. We report the performance of the resulting classifiers on three data sets of functional families from the Gene Ontology (GO) database. Our results show that NB($k$) classifier outperforms NB $k$-grams in terms of accuracy of classification (as measured by cross-validation) by a few percentage points. SVM $k$-grams outperforms NB($k$) in the majority of test cases. These results suggest the possibility of developing fully automated and computationally efficient approaches to construction of classifiers based on undirected graphical models of overlapping $k$-grams that can be easily updated as additional training data become available. Our results also show that further gains in accuracy of the classifiers are achievable (at the expense of increased computational demands and hence greater difficulty of frequent updates to the classifier as new training data become available) using SVM $k$-grams.

# 1 Introduction

Proteins are the principal catalytic agents, structural elements, signal transmitters, transporters and molecular machines in cells. Experimental determination of protein structure and function significantly lags behind the rate of growth of protein sequence databases. This situation is likely to continue for the foreseeable future. Hence, assigning proteins putative functions from sequences alone remains one of the most challenging problems in functional genomics [Eisenberg et al., 2000]. Improvements in annotating protein sequences can be expected to yield significant improvements in gene annotations. One class of sequence-based approaches relies on the comparison of the sequence in question to other sequences in a database of sequences with known function. Functional assignment is made by transference of function whenever sequences are sufficiently similar. A commonly employed notion of similarity is based on estimated sequence homology with programs such as BLAST and its derivatives [Altschul et al., 1997]. Sequence searches often return multiple hits, so significant human expertise is needed for interpreting results. The reliability of homologs detected by multiple sequence alignment rapidly drops once the pair-wise sequence identity drops below 30 percent [Rost, 1999]. A second class of sequence-based approaches for assigning putative functions to protein sequences rely on the detection of sequence patterns (Several automated tools for identifying conserved sequence patterns from a given set of sequences e.g., e-Motif and e-Matrix [Huang and Brutlag, 2001; Ben-Hur and Brutlag, 2003], MEME [Bailey et al., 1999] are available.) Motif databases can be queried using a protein sequence to obtain a list of conserved sequence patterns found in the sequence as well as functions associated with the respective patterns. The results can be used to assign putative functions to the protein sequence. In the case of protein families having sufficient numbers of well-characterized members, data mining approaches rooted in statistical inference and machine learning [Baldi and Brunak, 1998] offer an attractive and cost-effective approach to automated construction of classifiers for assigning putative functions to novel protein sequences. In essence, the data mining approach uses a representative training data set that encodes information about proteins with known functions to build a classifier for assigning proteins to one of the functional families represented in the training set (and if necessary, a default class indicating unknown function). The resulting classifier can then be used to assign novel protein sequences to one of the protein families represented in the training set after it has been validated using an independent test set (which was not used to build the classifier). Recent work by our group [Wang et al., 2003; Andorf et al., 2002] has explored the use of machine learning approaches to automated construction of such classifiers.

In this paper, we explore methods that use class conditional probabilities of $k$-grams ($k$-letter subsequences) to represent acid sequences. The first method uses a Naive Bayes classifier which treats each amino acid sequence as if it were simply a *bag* of amino acids. The second method (NB $k$-grams) applies the Naive Bayes classifier to a *bag* of $k$-grams ($k > 1$). Note that NB $k$-grams violates the Naive Bayes assumption of independence in an obvious fashion: neighbouring k-grams overlap along the sequence, adjacent $k$-grams have $k$-1 elements in common.

Our third method overcomes this problem by constructing an undirected graphical probabilistic model for $k$-grams [Charniak, 1993], which explicitly models the dependencies among overlap-

ping *k*-grams in a sequence. We train one such model per functional family. During classification, just as in the case of the Naive Bayes classifier, the sequence to be classified is assigned to the class that has the largest posterior probability given the sequence. We call the resulting classifier NB(*k*) to denote the fact that it models dependencies among *k* adjacent elements of sequences. Note that NB(1) is equivalent to NB 1-grams, which in turn is equivalent to the Naive Bayes classifier.

Our fourth method applies a support vector machine (SVM) [Boser et al., 1992; Vapnik, 1998] to classify amino acid sequences represented using class conditional probability distributions of *k*-grams in the sequence. SVMs have recently been applied successfully to many problems in computational biology including protein function classification [Lanckriet et al., 2003] and identification of protein-protein interaction sites from sequences [Yan et al., 2004]. However, to the best of our knowledge, previous work using SVM has not utilized a class conditional *k* gram probability-based representation of amino acid sequences.

While SVMs, unlike NB(*k*) and NB *k*-grams classifiers, do not have the advantage of training with only one pass through the training data, they are attractive in scenarios where higher accuracy of classification than that achievable by algorithms that make a single pass through the training data is desired. This increased accuracy comes at the expense of increased computational requirements - especially in cases where it is necessary to update the classifiers frequently as new training data become available. On a large data set a SVM classifier may take days to construct while NB(*k*) and NB *k*-grams can build a classifier in minutes. Hence, we explore an SVM that uses as input a *k*-gram class conditional probability distribution associated with the protein sequence to be classified. We call this third method SVM *k*-grams. This method is comparable to work using SVMs to predict subcellular localization based on amino acids [Hua and Sun, 2001; Bhasin and Raghava, 2004]. Their research focussed on using monopeptide and dipeptide composition. We consider larger ordered polypeptide composition in our study.

We compare NB *k*-grams, NB(*k*) SVM *k*-grams classifiers for assigning protein sequences to the corresponding GO (the Gene Ontology [Gene Ontology Consortium, 2000]) taxonomy of protein functional families. The sequence data sets used in our experiments were extracted from SWISSPROT [Boeckmann et al., 2003]. In our experiments, the NB *k*-gram classifier outperformed (in terms of classification accuracy), the standard Naive Bayes classifier by a large margin; the NB(*k*) classifier outperformed NB k-grams classifier by a few percentage points, and SVM *k* grams outperformed NB(*k*) in the majority of the test cases.

## 2 Method

In this section we outline the two probabilistic models we use for modelling the interactions among k consecutive elements in the sequence. First, we define a method to build a classifier associated with a probabilistic model.

**Classification Using a Probabilistic model**: Suppose we have a probabilistic model $\alpha$ for sequences defined over some alphabet $\sum$ (which in our case is the 20-letter amino acid alphabet).

The model $\alpha$ specifies for any sequence $\overline{S} = s_1,...,s_n$ the probability $P_\alpha(\overline{S} = s_1,...,s_n)$ according to the probabilistic model using the following (standard) procedure:

1. For each class $c_j$ train a probabilistic model $\alpha(c_j)$ using the sequence belonging to $c_j$.
2. Predict the classification $c(\overline{S})$ of a novel sequence $\overline{S} = s_1,...,s_n$ as given by:

$$c(\overline{S}) = \arg\max_{c_j \in C} P_{\alpha(c_j)}(\overline{S} = s_1,...,s_n)P(c_j)$$

Note that $P_\alpha(\overline{S} = s_1,...,s_n \mid c_j) = P_{\alpha(c_j)}(\overline{S} = s_1,...,s_n)$ have:

$$c(\overline{S}) = \arg\max_{c_j \in C} P_\alpha(\overline{S} = s_1,...,s_n \mid c_j)P(c_j)$$

**Naïve Bayes Classifier:** The Naïve Bayes classifier assumes that each element of the sequence is independent of the other elements given the class label. Consequently,

$$c(\overline{S}) = \arg\max_{c_j \in C} P_\alpha \prod_{i=1}^{n} P_\alpha(s_1 \mid c_j) \cdots P_\alpha(s_n \mid c_j)P(c_j)$$

Note that the Naive Bayes classifier for sequences treats each sequence as though it were simply a *bag* of letters. We now consider two Naive Bayes-like models based on $k$-grams.

**Naïve Bayes *k*-grams Classifier:** The Naive Bayes $k$-grams (NB $k$-grams) method uses a sliding a window of size $k$ along each sequence to generate a *bag* of $k$-grams representation of the sequence. Much like in the case of the Naive Bayes classifier described above treats each $k$-gram in the bag to be independent of the others given the class label for the sequence. Given this probabilistic model, the previously outlined method for classification using a probabilistic model can be applied. The probability model associated with Naïve Bayes $k$-grams classifier is as follows [Silvescu, 2004]:

$$P_\alpha(\overline{S} = [S_1 = s_1,...,S_n = s_n]) = \arg\max_{c_j \in C} P_\alpha \prod_{i=1}^{n-k+1} P_\alpha(S_i = s_i,...,S_{i+k-1} = s_{i+k-1} \mid c_j)P(c_j)$$

A problem with the NB $k$-grams approach is that successive $k$-grams extracted from a sequence share $k$-1 elements in common. This grossly and systematically violates the independence assumption of Naive Bayes.

**Naïve Bayes (*k*):** We introduce the Naive Bayes ($k$) or the NB($k$) model to explicitly model the dependencies that arise as a consequence of the overlap between successive $k$-grams in a sequence. Figure 1a) shows the dependency model for a sequence of 5 elements. We represent the dependencies in a graphical form by drawing edges between the elements that are directly dependent on each other. The graph for pair-wise dependencies is illustrated in Figure 1.b and the one for 3-way dependency is depicted in Figure 1.c

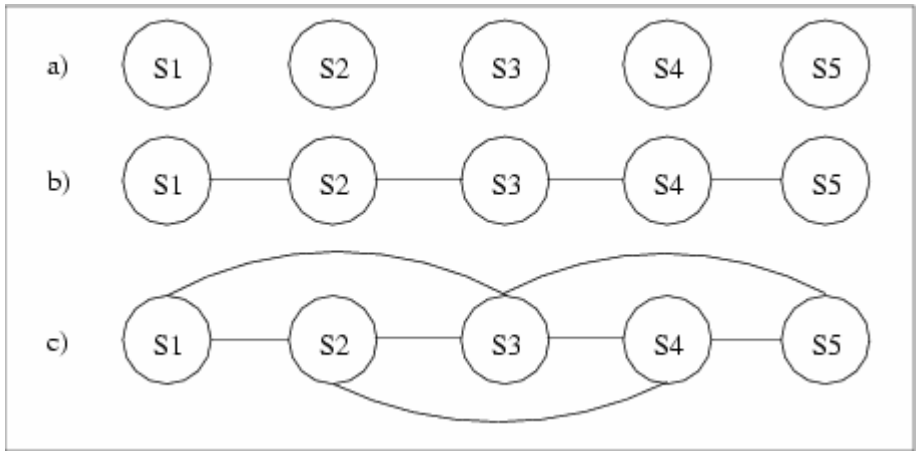


Figure 1: Graphical depiction of the dependence between the elements in a sequence of five elements using Undirected Graphical Models: a) shows the Naïve Bayes b) shows pairwise dependence (k = 2) and c) shows 3-way dependence (k=3).

Using the Junction Tree Theorem for probabilistic graphical models [Cowell et al., 1999], it can be shown that the correct probability model $\alpha$ that models the dependencies among overlapping *k*-grams is given by [Silvescu et al., 2004] for details):

$$P_\alpha(\overline{S} = [S_1 = s_1,...,S_n = s_n]) = \frac{\prod_{i=1}^{n-k+1} P_\alpha(S_i = s_i,...,S_{i+k-1} = s_{i+k-1})}{\prod_{i=2}^{n-k+1} P_\alpha(S_i = s_i,...,S_{i+k-2} = s_{i+k-2})}$$

Again, given this probabilistic model, we can use the standard approach to classification given a probabilistic model. It is easily seen that when $k = 1$, Naive Bayes 1-grams as well as Naive Bayes (1) reduce to the Naive Bayes model.

The relevant probabilities required for specifying the above models are estimated using standard techniques for estimation of probabilities using Laplace estimators [Silvescu et al., 2004].

**SVM *k*-grams:** Note that the NB(*k*) algorithm was developed because NB *k*-grams systematically violates the independence assumption of Naïve Bayes. Against this background, it is of interest to consider other methods that can utilize class conditional *k*-gram frequencies without relying on the independence assumptions made by NB *k*-grams and without the need for explicit modelling of dependencies as in the case of NB(*k*). Hence, we consider a Support Vector Machine (SVM) classifier which accepts as input, a class conditional *k* gram probability distribution for the protein (which is same as the one used by NB *k*-grams model) and outputs a class label. In this method, the SVM classifier can be seen as a second stage of a 2-stage classifier. The inputs to the SVM are supplied by the NB *k*-grams model.

## 3 Experimental setup and results

We compare the performance of the four classifiers - NB, NB *k*-grams, NB(*k*) and SVM *k*-grams on three protein function classification data sets.

**Data Sets:** The data sets used in this study are constructed as follows: First, a set of functional classes are chosen from the GO taxonomy of protein functional families. Then, the corresponding sequences are retrieved from the SWISSPROT data base [Boeckmann et al. 2003]. Because the GO taxonomy has the form of a directed acyclic graph, and many proteins are multi-functional, it is possible that a given protein belongs to multiple functional families. Hence, the resulting data set is filtered to remove proteins that had multiple GO class labels to ensure that the classes are non overlapping (i.e.., mutually disjoint) - a requirement of most standard machine learning and statistical methods for classification including the methods considered in this paper. (The development of principled approaches to classification of data that are labelled with multiple class labels is largely an open problem in machine learning).

The first data set was derived from families of yeast and human kinases. These families were chosen for this study because many of them are well-characterized, with known structures and functions. The data set used in this study consisted of 396 proteins belonging to the Gene Ontology functional family GO0004672, Protein Kinase Activity. We classified them according to the highest level below GO0004672. This consists of 5 groups. In GO, their labels are GO0004672, Protein Kinase Activity (102 proteins); GO0004674, protein serine/threonine kinase activity (209 proteins); GO0004713, protein-tyrosine kinase activity (69 proteins); GO0004712, protein threonine/tyrosine kinase activity (10 proteins); and GO0004716, receptor signalling protein tyrosine kinase activity (6 proteins).

The second data set is derived from two subfamilies of GO0003824, Catalytic Activity. This division is at a higher level of GO then the previous data set and consists of 376 proteins belonging to the Gene Ontology functional family GO0004672, Protein Kinase Activity (158 proteins) and GO001684, Protein Ligase Activity (218 proteins).

The third data set is a superset of data set two. It contains the Kinase data and Ligase data in addition to two other subfamilies of GO0003824, Catalytic Activity. These families are GO0004386, Protein Helicase Activity (110 proteins), and GO0016853, Protein Isomerase Activity (86 proteins). This data set tests the classifiers ability on a larger number of classes at a high level of GO and includes a total of 572 proteins.

**Experiments and Results:** The computational experiments were motivated by the following questions:

1. How do NB *k*-grams, NB(*k*), and SVM *k*-grams models compare with each other and against the baseline represented by Naïve Bayes (NB) classifier?
2. What is the effect of *k* (which can be viewed as a measure of the complexity of the models in question) on classification accuracy of the resulting classifiers?

NB *k*-grams and NB(*k*) models were constructed and evaluated on the three data sets for different choices of *k* from 1 to 4. Values of *k* larger than 4 were not considered because at higher values of *k* we run out of data to obtain reliable probability estimates. SVM *k* grams model, using a linear SVM kernel, was tested with values of *k* from 1 to 3. (Higher values of *k* were not explored because of computational and memory requirements). The reported accuracy estimates are based on stratified 10-fold cross validation. Within the 10-fold cross validation experiments, the majority of the individual standard deviations among classifiers were under 1% and never reached above 2%. This shows little variability among the classifiers used for these experiments.

| *k* | NB | NB *k*-grams | NB(*k*) | SVM *k*-grams |
|---|---|---|---|---|
| 1 | 66.1 | 66.1 | 66.1 | 84.1 |
| 2 | - | 81.3 | 88.6 | 90.7 |
| 3 | - | 89.9 | 92.7 | 90.3 |
| 4 | - | 90.4 | 91.6 | X |

Table 1: Kinase data set results.

| *k* | NB | NB *k*-grams | NB(*k*) | SVM *k*-grams |
|---|---|---|---|---|
| 1 | 77.9 | 77.9 | 77.9 | 97.6 |
| 2 | - | 83.5 | 84.6 | 100.0 |
| 3 | - | 84.0 | 85.6 | 100.0 |
| 4 | - | 85.9 | 90.7 | X |

Table 2: Kinase/Ligase data set results.

| *k* | NB | NB *k*-grams | NB(*k*) | SVM *k*-grams |
|---|---|---|---|---|
| 1 | 56.1 | 56.1 | 56.1 | 93.9 |
| 2 | - | 70.3 | 72.2 | 94.5 |
| 3 | - | 79.5 | 80.8 | 94.7 |
| 4 | - | 79.4 | 82.0 | X |

Table 3: Kinase/Ligase/Helicase/Isomerase data set results.

Table 1 – 3: All numbers represent classification accuracy estimated by 10-fold cross validation (Note: NB method applies only for $k$=1; SVM *k*-grams was found to be infeasible because of computational and memory requirements $k > 3$.

Because SVM is a binary classifier, and the problem calls for multi-class classifier, a separate SVM classifier was constructed for each class. The *i*th classifier is trained using the training data from class *i* as *positive* examples and the rest of the training data as negative examples. Note that unlike SVM, NB, NB *k*-grams, and NB(*k*) can build a single multi-class classifier.

Table 1 shows the results using the Kinase data set. We obtained a classification accuracy of 66% classification using Naive Bayes alone and an accuracy of 84% using SVM 1-gram. Increasing *k* to 2 resulted in significant improvements in accuracy: The accuracy increased to 81.3% for NB 2-grams, 88.6% for NB(2), and 90.7% for SVM 2-grams. In the case of NB(2) this represents 22% improvement over Naive Bayes and 7% improvement in classification accuracy over NB 2-grams. SVM on 2-grams only outperformed NB(2) by about 2% in terms of classification accuracy. NB 3-grams and NB(3) had accuracies of 89.9% and 92.7% respectively, with NB(3) (92.7%) actually outperforming SVM *3*-grams (90.3%). Increasing *k* to 4 resulted in little improvement on this data set. NB 4-grams improved by only 0.5% and NB(4), while performing better than NB 4-grams, has slightly worse accuracy relative to NB(3). This can be explained by the fact that as *k* increases, the probability estimates become less and less reliable (as we run out of data).

Similar results were obtained for the Kinase/Ligase and Kinase/Ligase/Isomerase/Helicase data sets. NB(4) outperforms NB(3) (by over 5% for the second data set [Table 2] and nearly 1.2% for the third data set [Table 3]) and NB 4-grams (nearly 5% [Table 2] and over 2% [Table 3]. SVM *k*-grams significantly outperforms NB(*k*), yielding 100% accuracy for the second data set and 94.7% accuracy for the third data set. This corresponds to a 14% improvement over the best accuracy of NB(*k*) on each of the data sets.

The experimental results demonstrate the superiority of both NB *k*-grams and NB(*k*) over Naive Bayes on all test cases using these datasets. Furthermore, in terms of accuracy, NB(*k*) outperforms NB *k*-grams, and SVM *k*-grams significantly outperformed NB(*k*) in terms of classification accuracy in two of the three test cases. In one of the test cases, the performance of SVM *k*-grams was comparable to that of NB *k*-grams. The results collectively demonstrate the utility of representing amino acid sequences in terms of class conditional probabilities of amino acids or *k*-grams of amino acids for sequence-based assignment of proteins to functional families.

## 4 Summary and Discussion

Development of robust methods for assigning putative functions to novel proteins and the discovery of sequence correlates of protein function are important challenges in bioinformatics.

This paper explored several methods for assigning protein sequences to functional families based on class conditional probability distributions of amino acids or short sub-sequences (*k*-grams) of amino acids on three data sets. The data sets were extracted from functional classes extracted from GO [Gene Ontology Consortium, 2000] and the corresponding sequences are extracted from SWISSPROT [Boeckmann et al. 2003].

Our results show that the NB (*k*) classifier, which models the dependencies among overlapping *k*-grams in a sequence, consistently outperforms NB *k*-grams and the Naive Bayes classifier in terms of classification accuracy. SVM *k*-grams, which also uses the class conditional *k*-gram

probabilities for the sequences outperforms NB(k) on two of the three data sets in terms of classification accuracy.

NB *k* grams and NB(*k*) require only one pass through the data which makes the resulting classifiers easy to construct and update as new data become available. In contrast, at present, there are no efficient algorithms for updating SVM classifiers to incorporate new data in an incremental fashion. This makes NB(*k*) an attractive alternative when using large data sets or data sets that are rapidly being updated or modified.

Some directions for future work include: exploration of classifiers constructed using reduced alphabet representations of protein sequence [Andorf et al., 2002]; development of principled approaches to assigning a protein sequence simultaneously to multiple classes (in the case of multifunctional proteins); incorporation of other sources of information (e.g., expression data, interaction data, structural features) to enhance the accuracy of function classification; examination of the resulting classifiers to identify testable hypotheses concerning sequence correlates of protein function and to guide the design of experiments to validate such hypotheses.

## Acknowledgements

This research was supported in part by grants from the National Science Foundation (0219699, 9972653) and the National Institutes of Health (GM066387).

## References

[Altschul *et al.*, 1997] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, W. Miller, and D.J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. In *Nucleic Acid Res.* Sep 1;2(17), pp. 3389 - 3402 ,1997.

[Andorf *et al.*, 2002] C. Andorf, D. Dobbs, and V. Honavar. Discovering protein function classification rules from reduced alphabet representations of protein sequences. In *Proceedings of the Conference on Computational Biology and Genome Informatics.* Durham, North Carolina, 2002

[Boeckmann *et al.*, 2003] B. Boeckmann, A. Bairoch, R. Apweiler, M. Blatter, A. Estreicher, E. Gasteiger, M. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider. The Swiss-Prot protein knowledgebase and its supplement trembl in 2003. *Nucleic Acid Res.* 31: pp. 365 – 370, 2003.

[Bailey *et al.*, 1999] T. Bailey, M. Baker, C. Elkan, and W. Grundy. Meme, mast, and metameme: New tools for motif discovery in protein sequences. *Pattern Discovery in Biomolecular Data. Oxford* University Press, Oxford, pp. 30 – 54, 1999.

[Baldi and Brunak, 1998] P. Baldi and S. Brunak. *Bioinformatics: The Machine Learning Approach.* Cambridge, MA: MIT Press, 1998.

[Ben-Hur and Brutlag, 2003] A. Ben-Hur and D. Brutlag. Remote homology detection: a motif based approach. *Bioinformatics* Vol. 19 Suppl. 1, 2003.

[Bhasin and Raghava, 2004] M. Bhasin and G. Raghava. ESLpred: SVM-based method for subcellular localization of eukaryotic proteins using dipeptide composition and PSI-BLAST. *Nucleic Acids Research*, Vol. 32, 2004.

[Boser et al., 1992] B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pp. 144 – 152, Pittsburg, PA, ACM Press, 1992.

[Charniak, 1993] E. Charniak. *Statistical Language Learning*, Cambridge: 1993. MIT Press, 1993.

[Cowell, et al., 1999] R. Cowell, A. Dawid, S. Lauritzen, and D. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, 1999.

[Eisenberg et al., 2000] D. Eisenberg, E. Marcotte, and T. Xenarios, and I. Yeates. Protein function in the post-genomic era. *Nature*. 405(6788): 823-6, 2000.

[Gene Ontology Consortium, 2000] The Gene Ontology Consortium. Gene ontology: tool for the unification of biology. *Nature Genet.*, (25) pp. 25 – 29, 2000.

[Hua and Sun, 2001] S. Hua and Z. Sun. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, Vol. 17 no. 8, pp. 721 – 728, 2001.

[Huang and Brutlag, 2001] J. Huang and D. Brutlag. The emotif database. *Nucleic Acids Res*. Jan 1:29(1): 202-4, 2001

[Lanckriet et al., 2003] G. Lanckriet, N. Cristianini, M. Jordan, and W. Noble. Kernal-based integration of genomic data using semidefinite programming. *In B. Schoelkopf, K. Tsuda and J-P. Vert (Eds.), Kernal Methods in Computational Biology*, Cambridge, MA: MIT Press, 2003.

[Rost, 1999] B. Rost. Twilight zone of protein sequence alignments. *Protein Eng*. 12(2): pp. 85 – 94, 1999.

[Silvescu et al., 2004] A. Silvescu, C. Andorf, D. Dobbs, and V. Honavar. Inter-element dependency models for sequence classification, Technical report, Department of Computer Science, Iowa State University, http://www.cs.iastate.edu/silvescu/papers/nbktr/nbktr.ps , 2004.

[Vapnik, 1998] V. Vapnik. Statistical learning theory. *Adaptive and learning systems for signal processing, communications, and control*. Wiley, New York, 1998.

[Wang et al., 2003] X. Wang, D. Schroeder, D. Dobbs, and V. Honavar. Automated data-driven discovery of protein function classifiers. *Information Sciences* 155: pp. 1 – 18, 2003.

[Yan et al., 2004] C. Yan, D. Dobbs, V. Honavar. A two-stage classifier for identification of protein-protein interface residues. *Bioinformatics* 20 pp. i371-378, 2004.