Chapter 1
A Hybrid Rule-Induction/Likelihood-Ratio Based Approach for Predicting Protein-Protein Interactions

Mudassar Iqbal, Alex A. Freitas, and Colin G. Johnson

Abstract We propose a new hybrid data mining method for predicting protein-protein interactions combining Likelihood-Ratio with rule induction algorithms. In essence, the new method consists of using a rule induction algorithm to discover rules representing partitions of the data, and then the discovered rules are interpreted as "bins" which are used to compute likelihood ratios. This new method is applied to the prediction of protein-protein interactions in the *Saccharomyces Cerevisiae* genome, using predictive genomic features in an integrated scheme. The results show that the new hybrid method outperforms a pure likelihood ratio based approach.

Key words: Data Mining, Bioinformatics

1.1 Introduction

Protein-protein interactions are involved in almost every cellular function, from DNA replication and protein synthesis to regulation of metabolic pathways [1]. Proteins interact with each other by physically binding themselves or with other molecules in the cell and form larger complexes to perform specific cellular functions. Hence, the study of protein-protein interactions is of utmost importance to understand their functions [7, 2], and detailed information about the interactions of proteins can have potentially very useful applications, e.g., predicting disease-related genes by looking at their interactions [28] as well as a potential use in developing new drugs that can specifically interrupt or modulate protein interactions [41]. Also, the study of these interactions at the genomic level can help understanding the large scale organization and features of the underlying network and the role of individual proteins within the network [46].

Centre for Biomedical Informatics and Computing Laboratory, University of Kent, Canterbury, U.K., {mi26,a.a.freitas,c.g.johnson}@kent.ac.uk

Consequently a number of experimental techniques for determining protein-protein interactions have been developed [39, 20, 12, 17]. Unfortunately the experimental determination of the interaction network of even very simple organisms is difficult and potentially erroneous, and the overlap among the interactions determined by different such techniques is very low [46, 42]. Hence, there is a clear motivation to develop new computational methods which can use data integrated from several genomic sources, as it is done in this work, as explained below. Many experimental and computational methods for the prediction of protein-protein interactions are discussed in recent reviews [36, 37, 41].

1.1.1 Computational Prediction of Protein-Protein Interactions

The purpose of computational methods is to predict unknown protein interactions using the relevant genomic information available, i.e., computational methods typically try to predict protein interaction by using data produced by other genomic techniques such as gene expression, localization etc, which are indirectly related to protein interactions. A variety of computational methods have been investigated for this problem so far. Many methods infer interactions from a single type of genomic data. For example, [3] and [4] address the question whether protein interactions can be predicted directly from the primary structure and associated data. Given a database of interacting proteins, they develop a machine learning system (Support Vector Machine) trained to recognise the potential interactions based solely on the primary structure and the associated physicochemical properties.

Another well-known method is called Rosetta Stone Method. In this method, Marcotte et al. [23] find and exploit a very interesting observation that:“some pairs of interacting proteins have homologs in another organism fused into a single protein chain (Rosetta stone”. Other biological hypothesis used for prediction of protein-protein interactions include similarity in phylogenetic profiles [11] and co-evolution of interacting partners [15, 16].

Another approach consists of casting the protein-protein interaction prediction problem as a type of combinatorial optimization problem (Satisfiability) by looking at the domain (conserved evolutionary units within the proteins) assignments of interacting and non-interacting protein pairs and then use a combinatorial optimization method to solve it. In [18] a particle swarm optimization method, a relatively new type of computational intelligence algorithm, was used to infer domain-domain interactions and then use the inferred domain-domain interactions to predict new protein-protein interactions. Yet another approach consists of analyzing protein-protein interaction data to infer domain-domain interactions using graph-theoretical belief propagation methods [19].

Also, there is a whole group of methods in which information from different genomic features is combined to predict interactions. Such methods are here called “Integrative Method”. For instance, in [45], authors build an integrative model using a kernel based method combining many heterogenous data sets and present a supervised learning approach for prediction of protein interactions. Jansen et al. in [21] formulate a Bayesian framework for combining different types of data and predict genome wide interactions in *Yeast*. The basic idea is that given certain features corresponding to protein pairs under consideration and their class attribute (interacting or non-interacting), one can estimate the likelihood of interaction for a given feature, and overall likelihood is estimated using a naive Bayesian formulation by assuming independance among all the features. Rhodes et al.[33] extend this Bayesian approach for predicting protein interaction to the human genome. The general idea of all the integrative methods is that one could combine various relatively weak features in a setting in which overall prediction is boosted by this integration of data. Some interesting observations are drawn in [22] regarding this data integration for protein interaction prediction. A detailed analysis of this data integration using different classifiers is researched in [5].

1.1.2 Overview of the Proposed Method

Our work is partly inspired by the work done by [21], in which they proposed a Bayesian method using the MIPS (Munich Information center for Protein Sequences [24]) complexes catalog as gold standard positive interactions, and a list of proteins in separate sub-cellular compartments as negative interactions, as there is no particular data set of experimentally determined non-interactions. They integrate multiple genomic data corresponding to protein pairs, including correlation in expression levels, functional similarity based measure, etc., as well as other experimental data about protein interactions, as predictive features for these positives and negatives. We use many of protein pairs features used in [21] and a subset of their gold standard non-interactions to conduct a data mining experiment in order to analyse the effect of hybridizing simple naive Bayesian style likelihood based method with some rule induction algorithms. Rule induction algorithms learn classification rules given the predictive features as well as the class attribute of a set of examples (protein pairs in this case). Those learned rules can be used to predict unknown protein interactions. We first analyze a simplified version of the naive Bayes classification method without using any prior information and analyze its behavior for different possible values of sensitivity and specificity of prediction. Then we combine that simplified naive Bayes formulation with another data mining algorithm, namely a rule induction algorithm which learns *IF-THEN* type classification rules from data.

In essence we propose a new hybrid approach where we use the partitioning of the data corresponding to the induced rules as “bins” from which likelihood ratios are computed and used to classify the data. We present a ROC curve analysis of results obtained using different threshold levels on the calculated likelihood values. Since these rules consist of multiple antecedents coping with attribute interactions, the bins defined by these rules should give us a better insight as compared to the uniform binning of attributes used in general naive Bayesian methods. We have applied this hybrid method for a specific biological application here, e.g., prediction of protein-protein interactions in the yeast *S. Cerevisiae* using multiple genomic features, but the underlying principles of the method are not application domain dependent, and indeed it can be applied to a wide range of classification problems in different application domains.

1.1.3 Organisation

The chapter is organized as follows. Section 2 discusses the background on rule induction algorithms for classification. Section 3 details the protein interaction data and data related to different arrtibutes used. Section 4 explains our method, starting with a brief introduction to Naive Bayes for classification and a rule discovery algorithm, the *PART* method for classification; and then proposes a new hybrid method combining features of both techniques. Comparative results based on the ROC analysis are presented in Section 5. Finally, Section 6 concludes the chapter.

1.2 Classification Rule Discovery Algorithms

Classification is one of the major data mining tasks. Given the data (examples) with the predictive attributes and class labels (e.g. interacting or non-interacting in a protein pair's case), the task of classification amounts to find relationaship(s) among the attributes and class labels. These relationships can be in the form of, for instance, *IF-THEN-ELSE* type rules, decision tree or conditional probabilities depending upon which approach is used for building the model [43]. A classification model is built using the training data, i.e., with class value known, and that learned model's quality is then tested on the test data, i.e., where the class value is absent.

Classification rules are one of the popular data mining approaches mainly because of their comprehensibility, by representing the gained knowldge in a form which is intutive to human understanding. These rules have two parts, i.e. the rule antecedent-which is a conjunction of multiple conditions on the predictor attributes and a rule consequent-which is the prediction of class attribute based upon the conditions in the antecedent. Conditons over individual attribute values potentially involve all relational operators.

$$IF cond_1 AND cond_2 ... THEN class \tag{1.1}$$

There are many approaches to built models involving rule sets for a classification problem. One most common and widely used approach is the separate-and-conquer approach, which we will discuss in some detail in the next section. Another popular classification method is the divide-and-conquer technique by building decision trees [31]. In the work below we describe a rule induction algorithm that uses aspects of both of these approaches. Therefore, we begin by reviewing the basic concepts of these two methods.

1.2.1 Separate and Conquer Approach

One of the two main approaches to rule induction is the *separate-and-conquer* approach. This approach was originally devised in [25] with the name *covering* strategy, whilst the term separate-and-conquer was introduced by Pagllao&Haussler [29]. Many different variants on this approach, designed to tackle different problems and data types, have been implemented; the review by Fürnkranz [10] gives an overview.

The general idea of the separate-and-conquer approach begins with an induction of a rule, via some rule induction algorithm, on the entire dataset. The examples that are correctly classified by this rule are then removed from the dataset, and the rule induction algorithm applied to this reduced dataset. An example is said to be correctly classified by a rule when the example's attribute values satisfy the conditions in the rule's antecedent and the example's class is the same as the class predicted by the rule's consequent. This process of rule induction and removal of covered training examples is repeated until the dataset is empty. In this way each example in the training data will be covered by at least one rule.

This approach has been used with a number of rule representations, i.e. the allowed structure of the antecedent in the *IF-THEN* rule. In early work on this approach [26] the antecedent of the rules is a simple relation between attribute and value; for example, a threshold for a numerical value. In other approaches, more sophisticated representations are allowed, for example in FOIL [32] PROLOG relations are used. A more sophisiticated approach is to allow the representation to expand when needed [40]; a number of approaches to this are reviewed by Fürnkranz [10].

Many different approaches have been used for the rule induction mechanism itself. These include both determininistic methods such as hill climbing [32] and beam search [27], and stochastic methods such as evolutionary algorithms [14].

One danger with these methods is that they can suffer from *overfitting*, where the model is fitted too specifically to the (noisy) training data set, and is therefore unable to generalise well to the test data set (unseen during training). Methods for tackling this problem revolve around the idea of *pruning* the rule set, either by removing whole rules, or by simplifying the precedent of the rule [35]. Such pruning methods fall into two main types: pruning methods that operate whilst the learning process is running (so called *pre-pruning* methods), and *post-pruning* methods that process the rule set after it has been generated. These methods are reviewed by Fürnkranz [10].

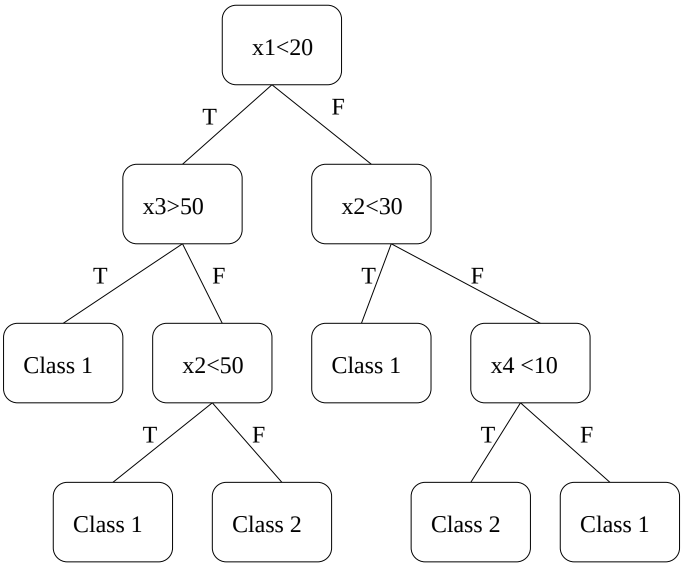

Fig. 1.1 An example of a decision tree.

1.2.2 Divide and Conquer Approach

In the previous section, the dataset was split by *instances*, each application of the rule induction algorithm removing some instances from the dataset. By contrast, the divide-and-conquer approach splits the attribute space as it works. The canonical representation used in the divide-and-conquer approach is the *decision tree*.

An example of a decision tree is given in figure 1.1. To predict the class of an unseen data instance using the tree, the tree is worked from the root. The algorithm evaluates the condition at the root node, and then moves on the the left or right child node depending on whether the result of evaluating that condition is true or false. This process is repeated until a leaf node is found; the leaf node names the class that should be assigned to that instance.

A number of methods have been devised for the induction of decision trees from data sets. The most widely used methods are those based on *information gain*, first introduced by Quinlan [30, 31]. This begins by constructing putative tree “stumps” [43], based on a number of options for the condition in the root node (how these options are constructed is algorithm and data-type specific). The training set is then distributed between the edges adjacent to this node based on this criterion, and a measure of the balance of classes associated with each of these edges is calculated. This measure is highest when an edge contains only one class (as there is no more decision to be made) and lowest when there is an equal balance of classes (as no information has been provided by the consideration of that condition). Based on this measure, the condition that maximises this information gain is chosen. This is then recursively repeated for lower levels of the tree, until one of the following conditions is satisfied: all classes are classified correctly (i.e. there are no “impure” edges); no more non-contradictory conditions can be created; or some algorithm-specific criterion for the simplicity of representation is satisfied (to avoid overfitting).

1.3 Protein Interaction Data and Predictive Features

We use four different features which were described as highly predictive features in the detailed analysis done by Lu et al. [22]. These features are explained below. All these features have been downloaded from supplementary material available with [22] and available online at `http://networks.gersteinlab.org/intint`.

- mRNA Co-expression (COE): Based on the hypothesis that interacting proteins have correlated expression profiles [21, 22, 13], this feature seems promising for the prediction of protein-protein interaction.
- MIPS Functional Similarity (MIPS): Interacting proteins often function in the same biological process [22]. The data associated with this feature was extracted from the MIPS functional catalog.
- GO Functional similarity (GOF): This data is based on a similar hypothesis as the MIPS data, but is created using the GO functional classification scheme. The details of preparation of the data are given in [21] and [22].
- Marginal Essentiality (MES): This is a quantitative measure of the importance of non-essential genes to a cell [47] and it is based on the Marginal-Benefit Hypothesis that many non-essential genes make a significant but small contribution to the fitness of the cell [38].

Of course there are many other genomic features available like essentiality, data derived from Rosetta stone method, etc; but most such features are very scanty, i.e., very few protein pairs have known values for these features, or their predictive power is very low as compared to the above mentioned highly predictive features. Hence, in this work we only use the above mentioned highly predictive features along with some high confidence gold standard interacting and non-interacting pairs of proteins.

We obtained the *S. Cerevisiae* protein interaction data from DIP (Data base of Interacting Proteins [34, 44]). We obtained nearly 5000 high confidence positive interactions in DIP, called CORE, which is a subset of the total number of reported protein interactions in DIP. Negative interactions are hard to find. As used by many researchers in this field we consider a protein pair as a negative example (i.e., the proteins in question do not not interact) if the proteins in the pair are not in the same cellular compartment [21, 22]. This gives us many hundred thousands of protein pairs which are not co-localized. As there are too many negative examples found in this way, we keep only a small subset of those. We obtained gold standard negatives from [22]. For both positive and negative gold standard data, we keep only those pairs with complete information, i.e., with no missing values in the four predictive features which we use. After this preprocessing we end up with 2122 positive interactions (positive examples) and 5656 negative ones in our gold standard set.

1.4 A New Hybrid Rule Induction/Likelihood-Ratio Based Method

In this section we will first discuss two different approaches which can be used for prediction of protein-protein interactions. First we will describe a naive Bayesian formulation which is based on the estimation of likelhood values of interactions given the predictive features, followed by a discussion of rule induction algorithms which output a classification rule set. Then we will describe a hybrid apprach which integrates both rule induction algotithms and likelihood ratios drawn from naive Bayesian approach.

1.4.1 From Naive Bayes to a Likelihood Based Approach for the Prediction of Protein-Protein Interactions

The Bayesian approach is widely used in inference problems in many different areas, including several types of bioinformatics problems. Jansen et al. [21] and Rhodes et al. [33] used a form of Naive Bayes classifier to predict protein-protein interactions by combining multiple features. Given a data set of interacting proteins,considered as positives, and a set of protein pairs separated in different cellular compartments, considered as negatives, prior odds are defined as:

$$O_{prior} = \frac{P(pos)}{P(neg)} = \frac{P(pos)}{1 - P(pos)} \tag{1.2}$$

Where $P(pos)$ and $P(neg)$ is the fraction of positives and negatives respectively among all pairs of proteins in the training data. The posterior odds that a pair of proteins interacts given the predictive features $f_1...f_n$ is:

$$O_{posterior} = \frac{P(pos|f_1...f_n)}{P(neg|f_1...f_n)} = O_{prior} * L(f_1...f_n) \tag{1.3}$$

$L(f_1...f_n)$ is the likelihood ratio and is defined as:

$$L(f_1...f_n) = \frac{P(f_1...f_n|pos)}{P(f_1...f_n|neg)} \tag{1.4}$$

Making the Naive Bayes assumption that the predictive features are independent from each other given the class (positive or negative), the likelihood ratio can be easily calculated as the product of individual likelihood ratios for each feature f_i as per Eq.1.5.

$$L(f_1...f_n) = \prod_{i=1..n} L(f_i) = \prod_{i=1..n} \frac{P(f_i|pos)}{P(f_i|neg)} \tag{1.5}$$

$L(f_i)$ is calculated as the fraction of positives having feature f_i divided by the fraction of negatives having feature f_i. As we are using a relatively small subset of total interactions and non-interactions and a reliable estimate of prior odds does not seem to be available, we do not use the prior odds at all in this formulation, and hence the posterior odds are the same as the likelihood ratio. Since the prior odds are not used, we analyze the predictive accuracy obtained for different threshold cutoffs on likelihood ratio values, instead. Hence, in this paper, we use a likelihood based approach for the prediction of protein interactions.

1.4.2 Generating Classification Rules for Protein-Protein Interaction Prediction

A popular type of data mining methods consist of building predictive models in the form of *IF-THEN* classification rules. More precisely, each rule has the form:

IF (condition(s) on attribute value(s)) THEN (class value)

Hence, each rule represents a relationship between the predictor attributes (features) and the goal attribute. Rules are discovered using the training set. The discovered rules are then used to predict the class value of examples in the test set, unseen during training [9]. Rule induction methods are known to present the knowledge discovered from the data in a comprehensible form to the users. Such comprehensible rules can be very helpful for the domain experts, for example biologists in our case, who can validate the discovered rules and potentially get new insight about the data. The discovered rules also have the potential to represent new knowledge about the problem at hand.

A variety of approaches exist for learning accurate and comprehensible rules from the data [43]. One line of research is to begin with building a decision tree and then transform it into a set of rules [31]. However, in the literature the term rule induction is often used to refer to an algorithm which discovers rules somewhat more flexible than a decision tree, in the sense that the discovered or induced rules cover data space regions that can have some overlapping (unlike the leaf nodes of a decision tree, which represent non-overlapping data space regions). Most rule induction algorithms use the previously discussed separate and conquer approach, which tries to determine the most powerful rule that underlies the data by sequentially adding conditions on the attributes to the rule, separates out those examples that are covered by the rule and repeats the procedure on the remaining examples [6].

For the problem at hand, we use a method called PART [8] for the classification of protein-protein pairs (examples or data instances) into interacting or non-interacting.

PART involves features of both decision tree building and rule induction algorithms-both of which were reviewed above. PART is available for use in the freely available data mining package WEKA [43]. The basic idea of this method is that it uses the separate and conquer strategy, as in the case of rule induction algorithms, in that it builds a rule, removes the examples it covers and continues creating rules for the remaining examples until none are left. But it differs from most rule induction algorithms in the way a rule is induced. To build a single rule, first a pruned decision tree is built for the current set of examples. Then the leaf with the largest coverage is made into a rule, and the tree is discarded. This process is iteratively repeated until all training examples are covered by the induced set of rules. The details about the PART method and its comparison to other competing methods are in [8].

1.4.3 Classification Rule Discovery as a Binning Method for a Likelihood-based Approach

When using the rule induction method described in previous subsection, most of the discovered rules contain conditions on multiple predictor attributes. For example, the following rule containing conditions on two attributes (GO and MIPS as defined in Section 2) and predicting class 0 (negative interaction).

$$\textit{IF (GO} \geq 3.85 \textit{ AND MIPS} \geq 5.45 \textit{ AND MIPS} \leq 6.15 \textit{) CLASS = 0}$$

There will be some negative examples satisfying this rule as well as some (perhaps small in number) positive examples. Unlike the Naive Bayes method, each discovered rule represents an interaction among the attributes in the rule's antecedent (since all attributes in that antecedent have to be satisfied, in order for an example to satisfy the rule). We can view each of these rules as a multiple-attribute binning of the data. This allows us to compute the likelihood in a way conceptually similar to Eq.1.4, i.e., the fraction of positive examples satisfying this rule antecedent divided by the fraction of negative examples satisfying the antecedent of this rule, but with the difference that, instead of computing a likelihood for each individual feature, we compute a likelihood for each "bin", i.e., each conjunction of the attribute values in a rule antecedent. Of course the bins and corresponding likelihoods are computed using rules discovered from the training data. We then evaluate this hybrid predictor's performance on the test set using a ROC curve. In other words, after having the rules, or these multi-attribute bins, we calculate their likelihood ratios and predictive accuracy by putting different thresholds on the minimum value of the likelihood ratio required to assign an example to the positive class. In this way we can analyze the whole range of threshold values like in the case of the Naive Bayes method, instead of the hard classification done by stand alone rule based method. We analyze the effects of these multi-attribute bins/rules against the assumption of the Naive Bayes method which assumes independence among the attributes given the class.

1.5 Results and Discussion

We present here a ROC (receiver operating characteristic) curve analysis of the results obtained using a likelihood-based approach as explained in section 1.4 and using our hybrid approach based on a rule learner combined with likelihood ratio test. A ROC curve graphically depicts the performance of a classifier at different levels of thresholds that we put on the minimum likelihood for prediction of positive interaction in this two class classification problem. In other words, for a given threshold value t, a test example (protein pair) is predicted to have interaction (positive class) if and only if the value of likelihood (Eq. 1.4) is greater than or equal to t. This kind of analysis gives us an opportunity to evaluate the classifier not just by the total number of classification errors it makes, but rather allows us to analyze what is the tradeoff among two different types of errors, i.e., false positive predictions and false negative predictions. It plots true positive rate (sensitivity) vs false positive rate (1-specificity), where each point in the curve belongs to a particular threshold on the likelihood value. In this way we can analyze the effect of different thresholds on predictive accuracy instead of analyzing the effect of a single threshold using prior odds.

We use the 10-fold cross validation procedure [43] in all experiments reported here. Both positive and negative interaction data along with the predictive features is divided into ten equal folds respectively. For each experiment, we divide the data (for both positive and negative classes along with their features separately) randomly in ten equal folds. Each time we use nine out of ten folds as training and the remaining one fold as a test. This process is repeated ten times, each time using a different fold as the test set. Likelihood values estimated during the training run are used to predict protein-protein interactions in the test examples. Sensitivity and specificity are defined by Eq. 1.6 and 1.7,

$$Sensitivity = \frac{TP}{TP+FN} \tag{1.6}$$

$$Specificity = \frac{TN}{TN+FP} \tag{1.7}$$

Where TP,TN,FP and FN are the number of true positives, true negatives, false positives and false negatives respectively. A ROC curve for a good classifier will be as close as possible to the upper left corner of the graph, with a large area under the curve. Fig.1.2 shows the ROC curves for pure likelihood-based approach (hereafter called LIKE) and the hybrid method (hereafter called LIKE-PART, i.e., Likelihood based classifier using PART for finding rules/bins). The corresponding areas under the curve are 0.8862 and 0.9325, showing a better predictive performance of the LIKE-PART hybrid.

We can see from the Figure 1.2 that taking into account the multi-attribute binning or the rules produced by the base rule learner has enhanced the overall performance of the classifier significantly, even though the features in this data are not so much correlated, as reported in [22]. Table 1.1 reports the results for the likelihood

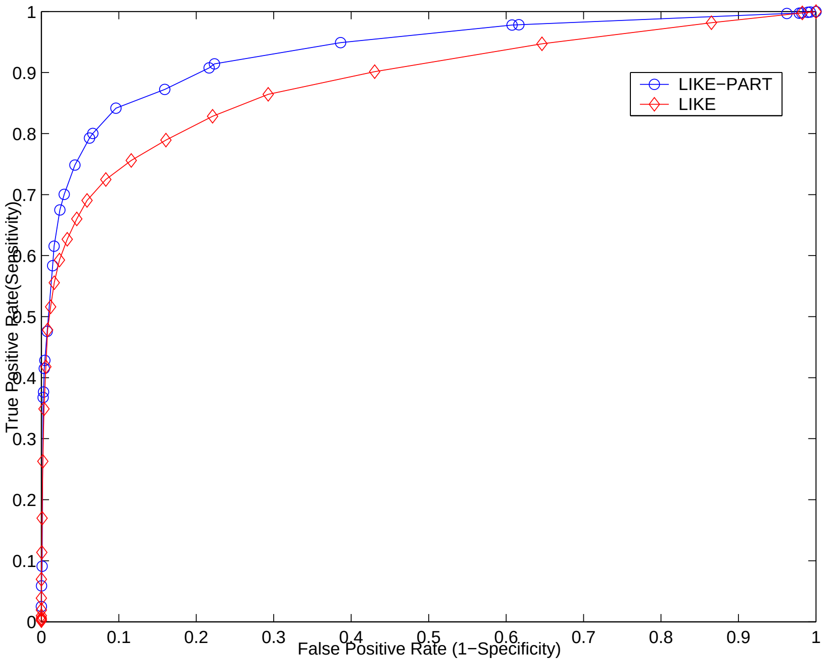

Fig. 1.2 ROC Curve for LIKE-PART and Pure Likelihood-based Approach (LIKE)

cutoffs which correspond to maximum predictive accuracies for both methods. A statistical significance test performed on the accuracy values over ten folds for these likelihood cutoffs gives a p-value of 0.0000017, which indicates that LIKE-PART outperforms the LIKE method very significantly.

Table 1.1 Results for maximum value of accuracy for both methods

$Method$	$LogLR_{cut}$	TPR	$FPR(1-Spec)$	$Sen*Spec$	Acc	TP/FP
LIKE-PART	0.49	0.748	0.0433	0.716	0.8998	6.48
LIKE	0.6	0.66	0.0457	0.63	0.874	5.41

1.6 Conclusions

In this work, we have addressed a challenging and important bioinformatics problem, namely the prediction of protein-protein interactions using a hybrid data mining technique combining rule induction methods with likelihood ratio based classifiers. We used integration of different genomic features for a small data set and implemented two versions of a likelihood ratio based classifier. We did not use any prior odds, but rather used only likelihood ratio and presented a range of results using ROC curve for different thresholds on the likelihood values used as a minimum value for the prediction of positive examples. We proposed a new hybrid method which used a known Rule Induction algorithm (*PART*) to induce rules from the training set taking into account possible attribute interactions and then interpret each rule as a bin for the likelihood based classifier. Since these bins were produced by taking into account attribute interaction, they avoid the unrealistic assumption of independence between attributes that is made by a pure likelihood based classifier. Then we compared the ROC curve of this new hybrid PART/Likelihood-based method with the ROC curve of the pure likelihood-based method and we observe that the hybrid method significantly improves as an overall classifier. Also, in the proposed method we can use different levels for likelihood value cutoff for final prediction, which gives us a more general setting where one can go for different levels of sensitivity and specificity.

We have evaluated this collaborative technique in the specific problem of predicting protein-protein interactions using genomic features, but the basic idea behind the technique, i.e., using induced rules as multi-attribute bins for the likelihood ratio based classifier, can be used for other classification problems easily, since it is independent from the application domain.

References

1. Alberts B. *et al.* (1989) Molecular Biology of the Cell, 2nd edn, *Garland*, New York.
2. Aloy P. and Russell R.B. (2006) Structural systems biology: modelling protein interactions. *Nat Rev Mol Cell Biol.* **7(3)**:188-97.
3. Bock J.R. and GoughD.A. (2001) Predicting protein-protein interactions from primary structure. *Bioinformatics***17(5)**,455-460.
4. Bock J.R. and GoughD.A. (2003) Whole proteome interaction mining *Bioinformatics* **19(1)**,125-135.
5. Browne F. *et al.* (2006) An assessment of machine and statistical learning approaches to inferring networks of protein-protein interactions. *Journal of Integrative Bioinformatics*,**3(2)**.
6. Cohen W.W., Fast effective rule induction. (1995) In *Proceedings of the 12th International Conference on Machine Learning, 115-123.*, Morgan Kaufmann.
7. Eisenberg D. *et al* (2000) Protein function in the post-genomic era. *Nature* **405(6788)**:823-826.
8. Frank E. and Witten I.H. (1998) Generating accurate rule sets without global optimization. In *Fifteenth International Conference on Machine Learning*, Morgan Kaufmann.
9. Freitas A.A. (2002) Data Mining and Knowldge Discovery with Evolutionary Algorithms. *Springer-Verlag*.
10. Furnkranz J. (1999) Separate-and-conquer rule learning. *Artificial Intelligence Review* **13(1)**:3-54.
11. Galperin M.Y. *et al.* (2000) Whos your neighbor?New computational approaches for functional genomics. *Nat. Biotechnol.*, **18**:609613.
12. Gavin A. *et al.* (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*,**415**:141-147.
13. Ge H. *et al.* (2001) Correlation between transcriptome and interactome mapping data from *Saccharomyces Cerevisiae.Nat. Genet.* **29**:482-486.

14. Giordana A. and Sale C. (1992) Learning structured concepts using genetic algorithms. In *Proceedings of the 9th International Workshop on Machine Learning*, D. Sleeman and P. Edwards (eds.), 169-178.
15. Goh C. *et al* (2000) Co-evolution of Proteins with their Interaction Partners. *J. Mol. Biol.*, **299**:283-293.
16. Goh C. *et al* (2002) Co-evolutionary Analysis Reveals Insights into ProteinProtein Interactions *J. Mol. Biol.*, **324**:177-192.
17. Ho Y. *et al.* (2002) Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature* **415**:180-183.
18. Iqbal M. *et al.* (2008) Protein Interaction Inference Using Particle Swarm Optimization Algorithm. In: Marchiori E. and Moore JH (Eds) EvoBio-2008, *LNCS*, Springer, 4973:61-70.
19. Iqbal M. *et al.* (2008) Message-Passing Algorithms for the Prediction of Protein Domain Interactions from Protein-Protein Interaction Data. *Bioinformatics*, doi:10.1093/bioinformatics/btn366.
20. Ito T. *et al.* (2001) A comprehensive two hybrid analysis to explore the yeast protein interactome.*PNAS* **98**:4569-4574.
21. Jansen R. *et al.* (2003) A Bayesian Networks Approach for Predicting Protein-Protein Interactions from Genomic Data. *Science*,**302**:449-453.
22. Lu L.J. *et al.* (2005) Assessing the limits of genomic data integration for predicting protein networks. *Genome Res.* **15**:945-953.
23. Marcotte E.M. *et al.* (1999) Detecting protein function and protein-protein interactions from genome sequences. *Science* **285**:751-753.
24. Mewes H.W. *et al.* (2002) MIPS:a database for genomes and protein sequences. *Nucleic Acids Res.* **30**:31-34.
25. Michalski R. S. (1969) On the quasi-minimal solution of the covering problem. In Proceedings of the *5th International Symposium on Information Processing (FCIP-69)*,**A3** (Switching Circuits), 125-128. Bled, Yugoslavia.
26. Michalski R. S. (1973) AQVAL/1—Computer implementation of a variable-valued logic system VL_1 and examples of its application to pattern recognition. In *Proceedings of the First International Conference of Pattern Recognition*, 3-17.
27. Michalski R. S. I. Mozetič, J. Hing and N. Lavrač (1986) The multi-purpose incremental learning system AQ15 and its testing application to three medical domains. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, 1041-1045.
28. Oti M. *et al.* (2006) Predicting disease genes using protein-protein interactions. *J. Med. Genet.* **43**:691-698.
29. Pagallo G., and Haussler D. (1990) Boolean feature discovery in empirical learning. *Machine Learning* **5**:71-99.
30. Quinlan J. R. (1986) Induction of decision trees. *Machine Learning*, **1**:81-106.
31. Quinlan J.R. (1993) C4.5: Programs for Machine Learning. San Mateo,CA, Morgan Kaufmann.
32. Quinlan J. R. and Cameron-Jones R. M. (1995) Induction of logic programs: FOIL and related systems. *New Generation Computing*, **13(3-4)**:287-312.
33. Rhodes D. R. *et al.* (2005) Probabilistic model of the human protein-protein interaction network. *Nature Biotechnology* **23(8)**:951-959.
34. Salwinski L. *et al.* (2004) The Database of Interacting Proteins: 2004 update.*NAR* **32**:D449-51.
35. Schaffer C. (1993) Overfitting avoidance as bias. *Machine Learning*, **10**:145-154.
36. Shoemaker B. A., Panchenko A. R.(2007) Deciphering ProteinProtein Interactions. Part-I: Experimental Techniques and Databases.*PLoS Computational Biology* **3(3):e42**.
37. Shoemaker. B. A., Panchenko A. R. (2007) Deciphering ProteinProtein Interactions. Part-II: Computational Methods to Predict Protein and Domain Interaction Partners, *PLoS Computational Biology* **3(4):e43**.
38. Thatcher J.W. *et al.* (1998) Marginal fitness contributions of non-essential genes in Yeast. *PNAS* **95**:253-257.

39. Uetz P. *et al.* (2000) A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae*. *Nature*, 403(1):623-627.
40. Utgoff P. E. (1986) Shift of bias for inductive concept learning. In *Machine Learning: An Artificial Intelligence Approach (Volume II)*, R. Michalski J. Carbonell and T. Mitchell, 107-148.
41. Valencia A. and Pazos F. (2002) Computational methods for the prediction of protein interactions. *Current Opinion in Structural Biology* **12**:368-373.
42. von Mering C. *et al.* (2002) Comparative assessment of large-scale data sets of protein-protein interactions. *Nature*, **417(6887)**:399403.
43. Witten I.H. and Frank E. (2005) Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco.
44. Xenarios I. *et al.* (2002) DIP: The Database of Interacting Proteins. A research tool for studying cellular networks of protein interactions. *NAR* **30**:303-305.
45. Yamanishi Y. *et al.* Protein network inference from multiple genomic data: a supervised approach. *Bioinformatics* **20(Suppl.1)**:i363-i370.
46. Yook S-H. *et al.* (2004) Functional and topological characterization of protein interaction networks. *Proteomics* **4**:928-942.
47. Yu H. *et al.* (2004) Genomic analysis of essentiality within protein networks. *Trends Genet.* **20**:227-231.