IMPROVING PROTEIN INTERACTIONS PREDICTION USING MACHINE LEARNING AND VISUAL ANALYTICS

By

MUDITA SINGHAL

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
Department of Computer Science

DECEMBER 2007

To the Faculty of the Washington State University:

The members of the Committee appointed to examine the dissertation of MUDITA SINGHAL find it satisfactory and recommend that it be accepted.

____
Chair

____

____

____

ACKNOWLEDGEMENTS

Reading the acknowledgements section in books was always a fun activity for me and now I know how difficult it is to pen your thoughts on everyone who had directly or indirectly contributed to an endeavor. But nonetheless I make an attempt to express my gratitude to all the people who contributed to my Ph.D. and thank them from the bottom of my heart.

To start with, I will like to express my sincere gratitude to my thesis advisor Dr. John Miller whose enthusiasm and consistent support played a major role in me pursuing and completing this thesis. He is a wonderful human being and I admire his articulacy and motivation which got me through most difficult problems in the past 5 years. I am also grateful to Dr. Haluk Resat with whom I journeyed the entire process of doing research and writing a research article. I learnt a whole lot from him and will always value it. Several other people contributed significantly to the development of my research interests and completion of this thesis. I sincerely thank Lars Kangas for introducing me to the field of machine learning and for several valuable discussions I have had with him. I thank Dr. Tjerk Straatsma for his support, Dr. Shira Broschat for being appreciative about my work and Ms. Kelly Domico for significant contributions to the public release of the CABIN software. I also thank Ms. Ruby Young for the helping me with the hundreds of questions I asked her over the course of these years.

Acknowledgements would not be complete without a mention of my dear husband Anuj for his confidence in me gives me the strength to go farther in life. Last but not the least I acknowledge all the support from my family because every phone call in which they asked “When are you completing your Ph.D.?” made me take a step forward towards completion.

I also acknowledge the funding for all this research which was obtained from the LDRD Program at the Pacific Northwest National Laboratory and was partially funded by the Advanced Scientific Computing Research, Office of Science, the U.S. Department of Energy. Pacific Northwest National Laboratory is a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy under Contract DE-AC06-76RL01830.

IMPROVING PROTEIN INTERACTIONS PREDICTION USING MACHINE LEARNING AND VISUAL ANALYTICS

Abstract

By Mudita Singhal, Ph.D.
Washington State University
December 2007

Chair: John H. Miller

The response of biological systems to external stimuli is ruled by their cellular interaction networks. This makes the problem of inferring cellular interaction networks essential to decipher the basic operational principles of biological systems. Knowing which proteins exist in a certain organism or cell type and how these proteins interact with each other are necessary for the understanding of biological processes at the whole cell level. The determination of the protein-protein interaction (PPI) networks has been the subject of extensive research and it has been shown that domain-domain interactions (DDIs) are good indicators of possible protein interactions, and can more accurately predict protein interactions than comparing full-length protein sequences. Despite the development of reasonably successful methods there is definite scope for improvement.

This thesis is aimed at developing machine learning based computational techniques that utilize domain information in the proteins to predict PPI networks. This research aims to make four major contributions to the field of PPIs. The first two are the development of two new PPI prediction algorithms, DomainGA and DomainSVM. DomainGA is a genetic algorithm based multi-parameter optimization method which quantifies DDIs and uses them to predict PPI. The second method, DomainSVM utilizes the DDI scores obtained from DomainGA in a Support Vector Machine (SVM) based learning system to improve PPIs prediction by overcoming the limitations of DomainGA. These two methods can be used as a two-step filtering process to validate experimentally detected PPI. The third contribution is score assignment to DDIs which is proven to be discriminatory between positive and negative PPI. Finally the fourth contribution is a visual analytic environment called CABIN (Collective Analysis of Biological Interaction Networks) which provides a one-of-its-kind tool to analyze, compare and integrate multiple predicted networks obtained from public data sources and/or inference algorithms such as DomainGA and DomainSVM. The predicted interactions accompanied by a confidence score and an exploratory visualization environment shall help researchers validate experimental observations and/or make an informed decision while generating hypothesis and models for designing new experiments.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Dedication

This dissertation is dedicated to my grand-dad (Daddy) for his unconditional love.

CHAPTER ONE

INTRODUCTION

Genome sequencing projects are producing large quantities of sequence data with the sequencing of more than 200 organisms to date, and still counting. Although sequence data lead to knowledge of proteins in the cell they provide little information about the biological processes involving these proteins. The fact that the sequence similarity of human and mouse genomes is between 70 - 90% suggests that species differences arise not because of the actual components themselves but because of how these individual components talk or interact with each other. This realization has led to rapid growth in the field of proteomics which aims at elucidating the structure, interactions, and functions of all the proteins within a cell of an organism. The study of proteomics in general and protein-protein interactions (PPIs) in particular will provide a more complete understanding of cellular processes and networks at the protein level. This can lead to a better understanding of disease mechanisms and aid drug target discovery by suggesting new means of intervention. This chapter is aimed at answering the following questions to provide a basic understanding of the importance of PPI networks and motivate the research reported in this thesis.

Why study protein interaction networks?

What are the existing experimental methods for detecting PPI?

Why develop computational methods for PPI prediction?

What are the existing computational methods for PPI prediction?

What are the objectives and contributions of this dissertation?

1.1. Why study protein interaction networks?

The interaction between proteins is fundamental to a broad spectrum of biological functions, including regulation of biological functions, metabolic pathways, progression through the cell cycle, and protein synthesis [1]. Although a complete understanding of protein functionality will require information on many levels such as knowledge of transcription, translational regulation, post-translational regulation, binding constants, and structures, answering basic questions, such as what proteins interact?, provides a foundation on which more complex regulatory information can be built [2]. Therefore, understanding PPI networks will have a significant role to play in the functional interpretation of fully sequenced genomes that have numerous genes of unknown function.

Studying protein interaction networks allows us not only to assess the role of individual proteins in the overall pathway but also to evaluate the redundancy of network components, identify candidate genes involved in genetic diseases, and set up the framework for mathematical models. For complex systems, the actual output may not be predictable by looking only at the individual components, but the complete picture is critical for correct biological assessment.

1.2. What are the experimental methods for detecting PPI?

The current techniques for detecting the proteins inside a cell involve separation via two-dimensional gel electrophoresis, followed by identification using tandem mass spectrometry [3]. There are several related high-throughput research methods for detecting PPIs such as the Yeast two-Hybrid (Y2H) (Uetz et al. [4] and Ito et al. [5] - first two comprehensive studies in yeast), affinity purification with mass spectrometric identification (Ho et al. [6] and Gavin et al. [7]), protein chips (Zhu et al. [8]), phage display (Tong et al. [9]), synthetic lethals (Tong et al. [10]), etc. The Yeast two-Hybrid (Y2H) and mass spectrometry techniques try to detect physical binding between proteins and have been widely used to detect PPI in different organisms.

The Y2H system detects the interaction between two proteins to be tested for interaction through an assay involving transcription activation of one or several reporter genes. As shown in Figure 1.1, the proteins are expressed as fusion proteins ('hybrids') in yeast: one protein is fused to a DNA-binding domain (BD), the other to a transcriptional activation domain (AD). Any interaction between them is detected by the formation of a functional transcription factor.

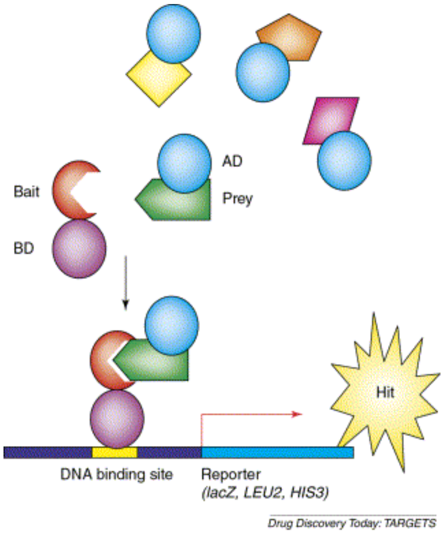


Figure 1.1: The Y2H system of detecting PPIs. The two proteins that physically interact with each other are observed by the formation of a transcription factor. Figure obtained from [11]

In the *mass spectrometry technique of purified complexes*, the protein of interest is tagged and inserted in a cell where it is expressed after forming a complex. The protein of interest and its complex is then isolated and then based on its mass-charge spectra; the proteins in the complex are identified. Von Mering et al. [12] gives a good background on other experimental approaches and their biases with respect to the distribution of interactions, functional categories of interacting proteins, and the drawbacks of these techniques.

1.3. Why develop computational methods for PPI prediction?

There are two main reasons for developing computational methods for PPI prediction: (1) to validate experimental observations of high-throughput experimental techniques because high-throughput methods are as much as 50% inaccurate and (2) to aid in the experiment design, obtaining complete interactomes of genomes computationally instead of using expensive, tedious, and highly inaccurate experimental methods since the interactomes of even well studied organisms such as Yeast are far from being complete.

As discussed by Legrain et al. [13] there are several flaws in experimental techniques such as Y2H for detecting PPI, namely a large percentage of observations are false positives (auto-activation, sticky prey) or false negatives (due to incorrect folding, inappropriate sub-cellular localization, absence of post-translation modifications). Due to this high number of false negatives in Y2H systems, the two exhaustive studies of the yeast proteome [4, 5] failed to identify as many as 90% of interactions previously described in the literature [5], and there was a very low overlap between these two exhaustive studies. Other experimental techniques also incur degradations due to self activators or promiscuous proteins as contaminants, protein concentration differences, and also lack of a benchmark. Analysis based on validation studies shows that only 30 – 50 % of high-throughput interactions are valid [14]. Legrain et al. [13]

discuss the various high-throughput techniques for detecting PPI, their general limitations, and the potential advances they make possible, especially when in combination with other functional genomics or bioinformatics analysis.

The second reason for studying PPI computations is the fact that whole genome PPI networks for well studied organisms are far from being complete with 30,000 estimated vs. 10,000 actual interactions in yeast, for example [12]. High-throughput experiments are tedious, expensive, and inaccurate and such experimental detections are not in place for most of the organisms that need to be studied to understand biological processes. However, it is essential to study interactomes of various organisms and to conduct cross-organism comparisons to be able to understand species evolution and biological processes.

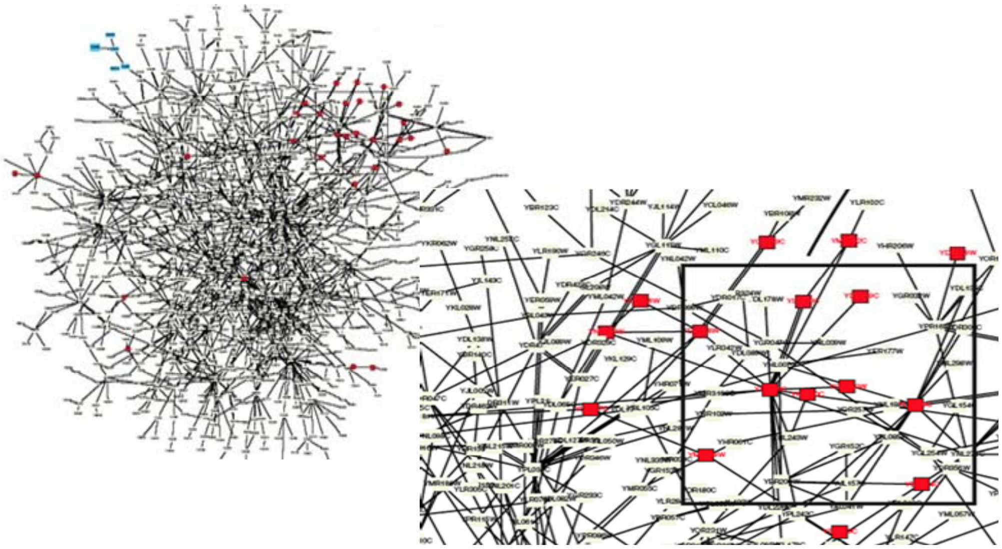


Figure 1.3: Complexity of interaction networks. Interaction networks are complex entities and there is a need for tools and algorithms to explore predict, and compare them. Figure obtained from Tucker et al. [2]

As shown in Figure 1.3 PPI networks are complex entities and there is a need to develop computational algorithms and tools to facilitate studying them in the context of validating experimental observations, partially replacing tedious, labor intensive, and potentially inaccurate [15] experimental techniques or for comparison of interactions obtained from several experimental or computational techniques. It is for these reasons that more and more research is focused on developing computational methods to predict whether two proteins interact or not. Complimentary in-silico methods capable of accurately predicting interactions would be of considerable value along with a need for tools to enable exploration, prediction, and comparison of PPI networks.

1.4. What are the existing computational methods for PPI prediction?

In the last decade extensive research has been conducted to develop computational techniques for determining PPIs based on genome sequence analysis [15-20], functional domain based approaches [21-31], and integration approaches [22, 32-36]. A good overview of genome sequence based approaches can be found by Valencia et al. [37]. Chapter 2 provides a survey of the methods proposed for PPI prediction in the last decade with a discussion on the advantages and limitations of these approaches.

1.5. What are the objectives and contributions from this dissertation?

The primary objective of this research is the effective utilization of structural domain information for predicting PPIs by developing methods that overcome the limitations and have better performance than existing methods. An added objective is to provide a tool to explore and analyze the predictions made using developed and existing computational and experimental methods for detecting PPIs. Inspite of a deluge of PPI prediction methods in the recent past, there is room for improvement in score assignment to domain-domain interactions

(DDIs) to be used for predicting PPIs. Most of the existing domain based methods do not accommodate the effect of multiple domain combinations, i.e., domains in one protein always functioning together to interact with other domains. Moreover, the accuracies achieved and assumptions made in most of the available predictive methods make it difficult to apply them in real world situations.

The goal of this research is to make four major contributions to the field of PPIs. The first two are the development of two new PPI prediction algorithms, DomainGA and DomainSVM. These two methods can be used as a two-step filtering process to validate experimentally detected PPIs. The third contribution is score assignment to DDIs which is proven to be discriminatory between positive and negative PPIs, and finally the fourth contribution is the development of a one-of-its-kind exploratory visual analytic tool called CABIN (Collective Analysis of Biological Interaction Networks) for comparing and analyzing multiple PPI networks obtained from a variety of predictive and/or experimental methods.

- Specific Aim 1: Develop and validate a genetic algorithm based methodology to quantify DDIs and use them to predict PPIs (DomainGA)
- Specific Aim 2: Develop and validate an SVM-based approach that effectively utilizes the DDI information by overcoming the limitations of DomainGA and assigns uncertainty scores to predicted PPIs (DomainSVM)
- Specific Aim 3: Develop an exploratory visual analytic tool to conduct exploratory analysis and comparison of experimental and/or computational predictions (CABIN)

An important advantage of computational analysis is its flexibility, high throughput ability, and low cost. In general it can be shown that the integration of the predictions obtained by the different computational approaches together with the experimental data can improve the functional assignment, as demonstrated for the S. cerevisiae genome [17]. Therefore, combining computational approaches and experimental methods will definitely accelerate the study of PPI networks, provide useful insights into the mechanisms of the biological processes in the organisms as well as enable us to better understand the complexity of life.

CHAPTER TWO

Existing Computational Methods for predicting PPI

Since the existing high-throughput experimental techniques for detecting PPIs are expensive and error prone; and the interactome of most organisms are far from complete as discussed in Chapter 1; several computational techniques have been developed in the last decade to complement experimental observations. The computational approaches proposed till date can be classified into three categories: 1) genomic inference based approaches 2) functional domain based approaches 3) integration approaches. This chapter reviews the proposed methods in literature for each of these categories.

2.1. Genomic inference based approaches

Several methods have been proposed that demonstrate the usefulness of computational techniques such as Artificial Neural Networks (ANN), Support Vector Machines (SVM), linear regression and random forests in deriving network topologies based on the primary sequence of the proteins. One of the first genome sequence-based approaches was proposed by Dandekar et al. [16] in 1998 called the *Gene Neighbor* method which looks at conserved operons across organisms to provide additional evidence that they are functionally coupled and are perhaps components of a protein complex or pathway. Around the same time Marcotte et al. [17] and Enright et al. [15] proposed the *Rosetta Stone/Gene Fusion* method. This method is based on the observation that two proteins expressed separately in one organism can be found as a single chain in the same or a second genome. This method infers PPIs from genome sequences given their observed homologies in other organisms, such that interacting proteins have fused into a single protein chain. The *Phylogenetic profiles* method was proposed by Pelligrini et al. [20] which uses the fact that co-evolved genes are more likely to be part of the same pathway or complex. This method uses co-occurrence or absence of pairs of non-homologous genes across genomes to infer functional relatedness. Bock et al. [18] presented a Support Vector Machines (SVM) based method for predicting PPIs based on the primary amino acid sequence and associated physiochemical properties such as charge, hydrophobicity and surface tension. The feature vector, the representation of a protein or protein pair in multi-dimensional space as required by a SVM, was formed as a concatenation of the above mentioned properties of Protein A and Protein B. They trained the SVM on PPI data from all organisms in the DIP database which is a curated database of exhaustive experimental studies such as that of [4, 5]. This method achieved a predictive accuracy of 80% in 10-fold CV tests and showed the potential of predicting PPI using sequence information. Bowers et al. [19] developed the *Prolinks database* which provides access to four genomic inferences based functional association methods namely the Phylogenetic profiles method [20], the Rosetta stone method [15, 17], a version of the Gene neighbor method [16] and the *Gene cluster method* which predicts functional linkage by using genome proximity. Martin et al. [38] extended the *signature descriptor* method for signature products to represent protein pairs and classified interactions in a SVM classifier. This approach is similar to that proposed by Bock et al. [18] but they bypass the transformation of the sequence information into physico-chemical properties of the protein by using a signature descriptor based representation. The drawback of this method is that it relies heavily on the completeness of the interaction map and on the quality of the functional annotations. These genomic inference based approaches are "*guilt by association*" based methods in which function is assigned to a protein by transposing existing annotations from its interacting partners.

2.2. Functional domain based approaches

One of the pioneering works of using domain interaction information to predict PPI was proposed by Sprinzak et al. [29] in 2001 called the *Association Method* in which they analyzed over represented sequence-signature pairs (domain pairs) among PPIs. The association method defines the fraction of interacting protein pairs among all protein pairs containing the domain pair as the measure of interaction between two domains. This method has the disadvantage that it assigns a high score to domain pairs that occur infrequently in positive PPI which might not be a correct assessment of the interaction probability. Wocjcik et al. [31] proposed a method to predict the protein interaction map of a target organism from the protein interaction map of a reference organism. The method is called the *Interacting Domain Profile Pairs (IDPP)* which is based on a combination of interaction and sequence data of one organism along with homology and clustering operations to obtain the interaction network of another organism. In simple terms it takes the set of PPI in one organism, transforms it to a set of DDI based on interaction and sequence information and uses that to predict PPI in another organism. They derive their own domain profiles instead of using a database such as pfam or Interpro that pre-defines relationships between proteins and domains. In 2002 the authors used the IDPP method to predict a virtual interaction map for Escherichia coli based on an experimentally derived map of Helicobacter pylori [30]. The drawbacks of this approach are that if some important DDI are missing or infrequent in the source organism (which is the case in our observations of occurrence comparisons of DDI in Human and Yeast for example) then

PPI predicted in the target organism will not be complete. Secondly this method relies heavily on the completeness, accuracy and level of detail (definition of protein domain) of the reference dataset. With these theoretical drawbacks and a lack of statistical cross validation analysis it is hard to estimate the applicability of this approach to a real world scenario. Deng et al. [22], present an optimization approach called the *maximum likelihood estimation* which infers domain interactions by maximizing the likelihood of the observed protein interaction data by optimizing it using the expectation-maximization (EM) algorithm. An EM method supplements observed data on PPI with missing data on DDI in an iterative manner to maximize the missing data parameters. They achieved 55.5% specificity and 55% sensitivity on their combined datasets, which are low accuracies for this method to be used for predictive purposes in a real world scenario. Gomez et al. [39] have developed *probabilistic models* for PPIs based on the frequency with which specific DDIs occur within known interactions. The domain interaction data is complemented with information on the topology of the network and is incorporated into the model by assigning greater probabilities to networks displaying more biologically realistic topologies. They use Markov chain Monte Carlo techniques for the prediction of posterior probabilities of the interaction between a set of proteins; allowing its application to larger data sets. In 2003 the authors extended this approach to incorporate negative PPI by describing an attraction-repulsion model in which the interaction between a pair of proteins is represented as the sum of attractive and repulsive forces associated with small domain sized features along the length of each protein [24]. Kim et al. [27] improved the association method [29] by using the number of domains in each protein to accommodate for the correct probability assignment to infrequent domains. Recently, Riley et al. [28] have presented the *domain pair exclusion analysis* method for inferring domain interactions from multiple organisms using the Database of Interacting Proteins (DIP) [40-43]. These methods provide evidence to support the fact that proteins interact via physical units such as evolutionary conserved protein domains (e.g., as defined in the PFAM database [44-46]). Guimaraes et al. [25] present a *parsimony-driven* approach where domain interactions are inferred using linear programming optimization and false positives in the protein network are handled by a probabilistic construction. The advantage is that they do not rely on PPI networks. These approaches do not consider the fact that multiple domains in a protein can interact with multiple domains in another and the possibility of a domain pair appearing both an interacting and a non-interacting protein pair. Proteins typically contain two or more domains; given that about two-thirds of proteins in prokaryotes and four-fifths in eukaryotes are multidomain proteins. Therefore since these methods assume independency of DDIs they suffer from the general limitation of the association method which is ignoring other DDI information between the protein pairs.

Chen et al. [21] propose a domain-based *random forest* of decision trees to infer protein interactions. It takes into account combinations of domains by exploring all possible domain interactions and making predictions based on them unlike the previous methods. They tested their system on yeast with equal number of positive and negative samples and achieved sensitivity of 80% and specificity of 65%. Their feature vector representation is a vector of domains with a value of zero, one or two based on if the domain is present in none, one or both the proteins. Unfortunately this representation does not provide information on if one or more domains were from the same protein or from two different proteins. Han et al. [26] proposed a domain combination based method which considers all possible domain combinations as the basic units of each protein. The domain combination interaction probability is also based on the number of interacting protein pairs containing the domain combination pair and the number of domain combinations in each protein. The method considers the possibility of domain combinations appearing in both interacting and non-interacting sets of protein pairs. Although these are promising methods, most of these methods are tested on an equal number of positive and negative examples and as discussed by Ben-Hur et al. [47], that assumption can severally limit the usefulness of a prediction method to a real world scenario in which there is a huge imbalance between positive and negative interactions.

2.3. Integration approaches

The use of various intersecting data types [12, 14, 34, 40, 48-52] has also been proposed to improve the PPI networks. Dohkan et al. [33] show that SVM is a robust technique to predict PPI pairs. Using interfacial surface characteristics such as amino acid compositions, hydrophobic moments, hydrophobicity, and protein length along with the domain information about the proteins; they have shown that their approach is able to identify more positive PPIs than the Maximum Likelihood Estimation technique used by Deng et al [22]. Jansen et al. [34] in 2003 used a Bayesian integrative approach to assign confidence to PPI based on experimental interactions data obtained from Y2H and mRNA expression and integrated it with genomic features such as GO biological processes and MIPS function. This was a seminal paper showing the usefulness of multiple feature integration in predicting PPI. In 2005 Lu et al. [35], showed with the use of 16 features that there is a limit to genomic data integration beyond which the predictive power of an approach does not increase. The reason for the lack of increase in the predictive power is however the unavailability of high-coverage genomic features. Ben-Hur et al. [32] use a SVM classifier to predict PPIs using sequence based information such as K-mer compositions [53] and PFAM domains [46] and non-sequence based information such as GO (Gene Ontology) annotations [54], homology and mutual clustering coefficient. They propose a pair-wise kernel as a measure of similarity between pairs of proteins. Experimenting with different kernel transformations, their results indicate that a linear kernel is best to represent sequence and non-sequence based information. They incorporated the knowledge about the reliability of the PPIs into the training procedure using the SVM soft-margin parameter *c*. This parameter puts a penalty on patterns that are misclassified or are close to the SVM decision boundary. Their negative set selection was based on random interactions between proteins in the positive set and they chose an equal number of negative samples as the positive samples, which is not representative of a real world scenario. Ng et al. [36] devised an integrative approach to computationally infer protein domain interactions and showed that the use of heterogeneous data sources with domain evidence improved protein interaction detection sensitivity. They have developed a database of putative DDIs called *InterDom* [55] which differs from the above mentioned statistical methods by attempting to directly quantify the strength of the DDIs. In the InterDom database, the DDIs are derived by combining information from multiple sources: domain fusions, protein interactions and complexes, and scientific literature. A probability-based scoring scheme is used to assign higher confidence to domain interactions that are derived independently by multiple methods from different data sources [36, 55]. Although it is a novel effort, our analysis results show that the InterDom scores need significant improvements before they can be used for predictive purposes for PPIs as discussed in detail in the chapter three.

2.4. Summary

As quite a few of the methods discussed above suffer from low accuracies there is a definite opportunity for improvement. Secondly, although some of these methods claim a high accuracy, their method of selection of negative interactions introduces some bias. They randomly select a subset of the negative samples which is equal to the positive sample size. But in a real world scenario because of the size difference between negative (non-interacting pair) and positive (true interaction) PPIs even a very low false-positive prediction rate can result in a situation where most of the predicted interactions are incorrect [34]. Thirdly, most of these methods do not achieve high sensitivity (recall) value, which as discussed by Marina et al., is an important measure for testing the performance of a bioinformatics approach. The methods developed in this thesis outperform the existing approaches with the right assumption of more non-interacting protein pairs than interacting protein pairs. Moreover these methods use only the fundamental domain information and therefore they can be applied across organisms.

CHAPTER THREE
Benchmark Data selection and processing

The quality of the available training data is very important to the success of a learning algorithm-based method. For the PPI network construction studies, the training dataset ideally contains a list of truly positive interactions (i.e. real PPIs) and a list of non-interacting pairs of proteins (i.e. negative examples) called the gold standards for a particular organism. A well-performing interaction scoring scheme should have predictive power and be able to discriminate between the true and false observations.

3.1. Benchmark Data

Since definite identification of true and false PPIs is problematic due to the reported inaccuracies in the experimental methods, one has no choice but to create positive and negative PPI lists and assume that the list is correct. Fortunately, there are efforts devoted to construct PPI lists for yeast that are as reliable and correct as possible [34, 38, 50]. The training and test datasets selected for this research are obtained from these earlier compilations. For the positive and negative yeast PPIs, the information available at the Munich Information Center for Protein Sequences (MIPS) [56, 57] in the version originally compiled by Jansen et al. [34], which contains 8250 positive and ~2.7 million negative PPIs is utilized. This set of PPIs represents the interactions between proteins that are present in the same complex. Negative PPIs were obtained by using the protein location information by assuming that proteins residing in different sub-cellular compartments do not interact. Although this assumption is not entirely valid (as discussed by Ben-Hur et al. [47]), because of the low error rate, its effect on the outcome of the prediction algorithms is assumed to be unimportant.

The domain information is obtained from the InterPro database [58, 59] which uses sequence alignment and regular expression based patterns as well as profiles with complex probabilistic scoring mechanisms such as hidden Markov Models (HMMs) to identify domains in a protein. InterPro capitalizes on the individual strengths of a number of databases including PROSITE [60], Pfam [46], PANTHER [61], and PRINTS [62] as well as sequence-cluster based methods such as PSI-BLAST [63] on well-characterized proteins to derive protein domains. As it unifies Pfam with other databases, use of the InterPro database allowed us to obtain a better coverage for the domains of the proteins of interest.

Finally the cross-verification tests for DomainGA scores were performed using the Core and Full yeast PPI datasets from Uetz et al. [64]. The Core subset of DIP contains the pairs of interacting proteins (*ScereCR20060402* list downloaded on 04/02/2006) identified in the budding yeast that were validated according to the criteria described in Deane at al. [65]. The Full Yeast set corresponds to the subset of DIP that contains all the pairs of interacting proteins identified in the budding yeast (*yeast20060402.lst* file downloaded on 04/02/2006). These sets contain 5952 and 17471 positive PPIs, respectively. The datasets originally compiled by Rhodes et al. [40] were used in the across-organism cross-verification test with the human PPIs in DomainGA. These sets were obtained from the Human Protein Reference Database (HPRD), and the lists contain 364,645 positive and ~40 million negative PPIs.

3.2. Data pre-processing

One of the biggest challenges with biological data in general and the PPI datasets obtained herein is identifier mapping. There are multiple identifiers such as Uniprot-Accession, IPI number, GI number, Entrez Gene-Id, Common Names, and Gene Symbols etc. that are used to represent a biological entity such as a protein/gene. Very few of the public databases agree on the usage of the same set of identifiers to represent the entities. This leads to the problem of mapping between identifiers to be able to compare entities across databases. The publicly available mapping information is redundant and incomplete to say the least. More often than not multiple identifier mappings are required to get from one set of identifiers to the other. For instance: pathway maps available on websites such as CancerCell [66] and Human Protein Reference Database (HPRD) [67] represent proteins using common-names/gene-symbols which need to be compared to a domain information based prediction system that uses uniprot-ids since the domain information available from the Interpro [58, 59] database is valid only for uniprot-ids; while the gold standards data from a database such as HPRD defined PPI using entrez-ids. The problem is exacerbated by the fact that there can be one-to-many relationships between these identifiers (For example: One IPI number generally has more than one equivalent uniprot-ids/accession numbers). There are no well defined criteria for resolving the inconsistencies and redundancy of these identifiers in an automated manner.

These problems are dealt with the creation of equivalency mappings between proteins based on their domain compositions. Since these are domain based approaches and a protein is represented by a string of domain-ids, if two proteins come from the same parent identifier and contain the same domains, it is treated as one protein for this purpose.

3.3. Subsets creation

To avoid over-fitting not all domain-domain interactions in an organism are selected as the parameter set used for quantification in the DomainGA method or as elements in the DomainSVM feature vector. A subset of the DDI is selected which requires sub-setting the PPI training data. Therefore these PPI datasets were further processed to obtain the relevant subsets (Table 3.1) for use as training data in both DomainGA and DomainSVM. The training and testing datasets have been created in two different ways, by introducing two new concepts. The *closed set* is a subset of the PPIs such that all the domain-domain pairs in the included PPIs are included in the parameter set. All the other PPIs are not selected. In contrast, the PPIs in which the involved protein pair has the potential to interact through one or more of the domain-domain pairs not included in the parameter set are included in the *inclusive set*. In other words, PPIs in the inclusive set may interact through the domain-domain pair that is chosen in the parameter set, but these PPIs may have other DDI pairs that are neglected (not included in the chosen parameter set). Details of the resulting set sizes used in training the DomainGA and DomainSVM are reported in Table 3.1.

No of Parameters / Data Set [a]		PPI	Retained Interactions
103	Inclusive	Positive	342
		Negative	14,402
867	Inclusive	Positive	1,882
		Negative	79,413
344 [a]	Closed	Positive	435
		Negative	3,139
2466	Inclusive	Positive	2,308
		Negative	162,115
1216 [a]	Closed	Positive	734
		Negative	13,146
5095	Inclusive	Positive	2,666
		Negative	243,866
3060 [a]	Closed	Positive	1,448
		Negative	25,651

Table 3.1. Details of the yeast MIPS datasets used in the studies. Starting yeast dataset was obtained from Munich Information Center for Protein Sequences (MIPS) site [57] and it contained 8250 positive and ~2 million negative PPIs [34]. Retained interactions column report the number of entries for the sets after the original dataset is filtered according to the domain pairs included in the parameter set. Further details can be found in the Methods section.

a These are the closed set versions of the 867, 2466, and 5095 parameter inclusive sets. It is important to note that during filtering to obtain the closed PPI sets, occurrence of some of the domain pairs are nullified and these parameters cannot be truly optimized during the GA runs. So these closed sets are a subset of their corresponding inclusive sets.

CHAPTER FOUR
DomainGA

Abstract

This chapter presents the DomainGA which is a multi-parameter optimization method in which the available PPI information is used to derive a quantitative scoring scheme for the domain-domain pairs. The scores are then used to predict whether a pair of proteins interacts. This chapter discusses the design of the DomainGA methodology and its statistical validation with respect to the selection of the parameter sets to be optimized, score ranges, and fitness evaluation rule. DomainGA method surpasses other existing methods by achieving very high explanation ratios for the positive and negative PPIs in model organism Saccharomyces cerevisiae. Cross-verification tests were conducted on human PPIs and the scores of the optimized parameters were compared with structurally observed domain interactions obtained from the iPFAM database which shows that the DomainGA method holds great promise to be applicable across multiple organisms. Since DomainGA uses fundamental domain information, it can be used to create potential PPIs and the erroneous predictions can be filtered further using supplementary approaches such as those based on literature search; DomainSVM method discussed in Chapter 5; or other predictive methods discussed in Chapter 2.

4.1. Introduction

DomainGA advocates a quantitative approach that uses the structural domains of proteins as a fundamental filtering step in inferring biological PPI networks. The underlying premise is that proteins interact with each other through their smaller substructures (i.e., domains), which have the biophysical properties that are instrumental in protein-protein complex formations [68]. The validity of this assumption stems from the fact that evolutionarily conserved polypeptide domains can be thought of as structural building blocks that define and regulate the functionality of the proteins. Such ideas also form the foundation of the databases, such as the Pfam database [46], allocated to the characterization of protein domains.

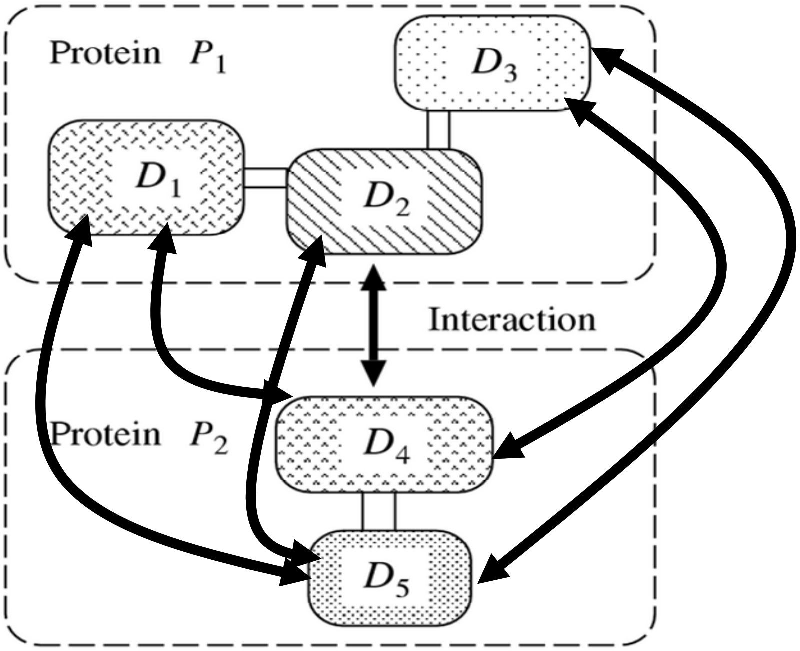


Figure 4.1: Protein-Protein Interactions and the role of domains. Proteins P1 and P2 are composed of domains D1, D2, D3 and domains D4, D5 respectively. The knowledge of proteins P1 interacting with P2 can be interpreted as and/or combinations of domains interacting in the two proteins.

In the domain-based structural quantification approach, the knowledge about the strength of the interaction between domain d_i in protein X and domain d_j in protein Y is used to predict whether proteins X and Y interact as shown in Figure 4.1. Corollary to this would be that for a given list of PPIs, can the possible interactions (and their strengths) between domain pairs be determined? This idea was behind the development of the InterDom database and domain interaction scores [36, 55], and it forms the starting point of the study presented here. DomainGA first develops a scoring scheme for the interactions between the *functional domains* of the proteins and then uses it to predict the strength of interaction between *protein pairs*.

In the InterDom database, the DDI are derived by combining data from multiple sources: domain fusions, protein interactions and complexes, and scientific literature. A probability-based scoring scheme is used to assign higher confidence to domain interactions that are derived independently by multiple methods from different data sources [36, 55]. These scores can be used to evaluate the InterDom method in terms of its predictive power of the PPIs. If the domain interaction scores have good discriminatory power, predicted PPI scores for the positive and negative PPI lists should be different - at least qualitatively. However, as shown in Figure 4.2, score distributions for the negative and positive lists for the human and yeast PPIs have considerable overlap. It should be noted that this analysis overlooks certain factors that are also determinants of domain-domain, and therefore protein-protein, interactions. Subtle differences in actual domain structures such as the ones due to amino acid composition, environmental factors, and whether the placement of the domain is in an accessible portion of the protein would be a few of such factors. For these reasons, the lack of a clear separation between the scores for the positive and negative PPI predictions may not be entirely due to the InterDom scores. Therefore, not a complete but only a reasonable separation between the curves is to be expected in Figure 4.2. However, the observed large overlap clearly indicates that there is room for improving the InterDom DDI scores, if they are to be used to predict PPIs in this manner.

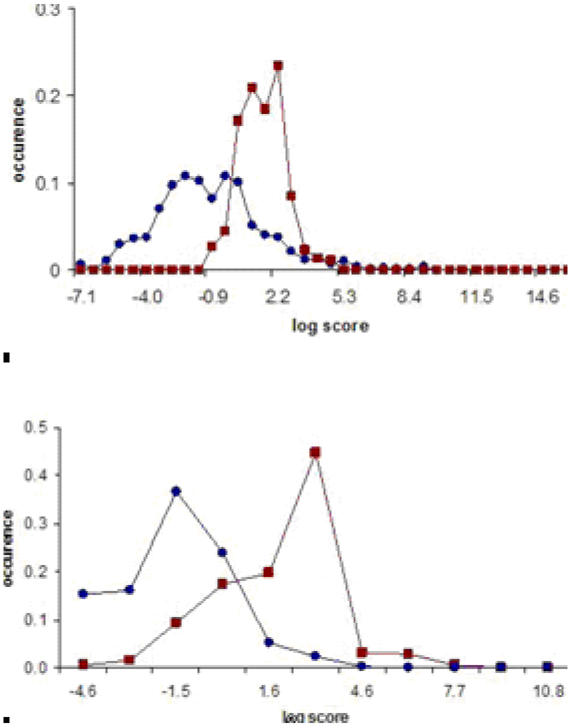


Figure 4.2: Comparing positive and negative PPIs computed using InterDom DDI scores. Comparison of the strengths of the positive (red line with squares) and negative (blue line with circles) PPIs computed using the InterDom DDI scores. The interactions with a score of zero are not reported. The histogram curves were calculated by binning the logarithm of the PPI scores that were computed using the maximum-score detection rule. Vertical axis shows the percentage of the PPIs with interaction scores that are within the strength interval of a particular bin. Top: Yeast PPI obtained from Munich Information Center for Protein Sequences (MIPS); Bottom: Human PPI obtained from Human Protein Reference Database (HPRD).

DomainGA quantifies the protein DDI by optimizing them as parameters in a genetic algorithm. The algorithm generates a set of DDI scores, which are then used to classify the interactions into three categories: high, low, and fuzzy. As in any machine-learning technique, DomainGA approach requires good-quality training data. Since a large quantity of high quality public data exists, *Saccharomyces cerevisiae* (yeast) is arguably the best model organism for testing new algorithms. Therefore the algorithm is benchmarked using the PPI data available for *S. cerevisiae* with details provided in Chapter 3.

4.2. Research Design and Methodology

Figure 4.3 depicts the flow of logic in the DomainGA methodology which is discussed in detail in this section. The actual DomainGA algorithm is written in C but it is supported by several Python scripts required for preprocessing the input data and post-processing the results. The details of the benchmark data and its pre-processing to circumvent several challenges such as obtaining negative interactions; dealing with conflicting data in positive and negative PPI; resolving identifier induced redundancy; and appropriate sub-setting of the training data & parameter set to avoid over-fitting the problem space are discussed in Chapter 3.

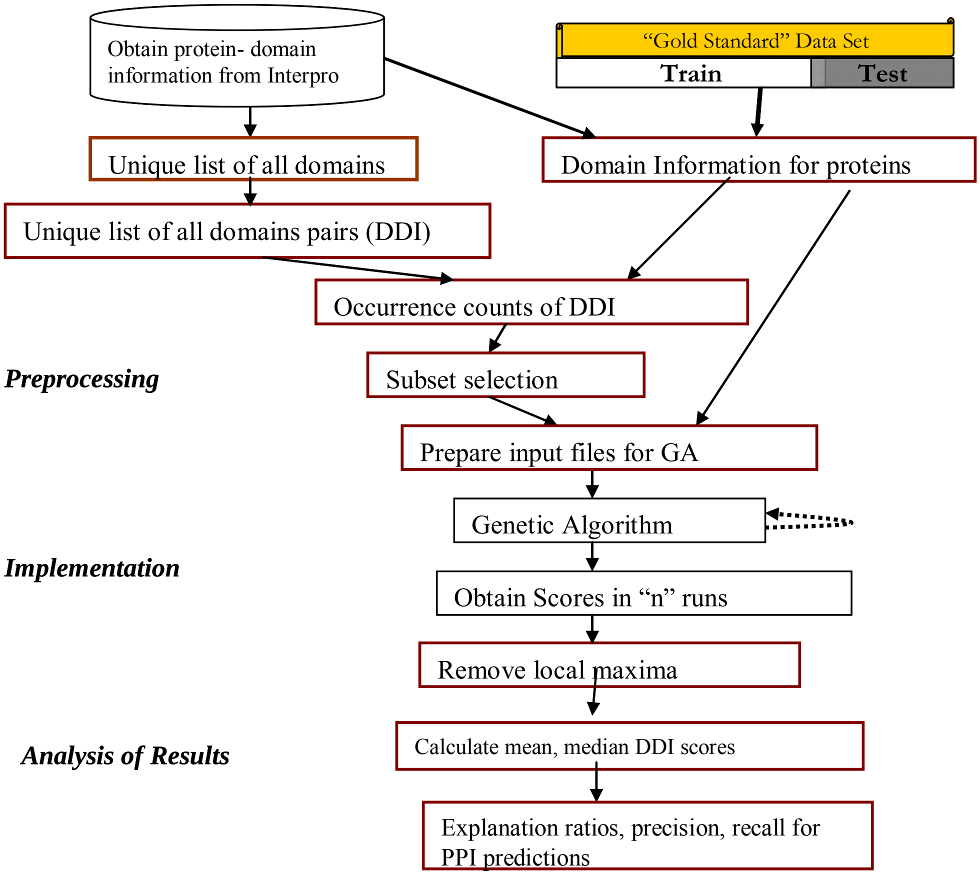
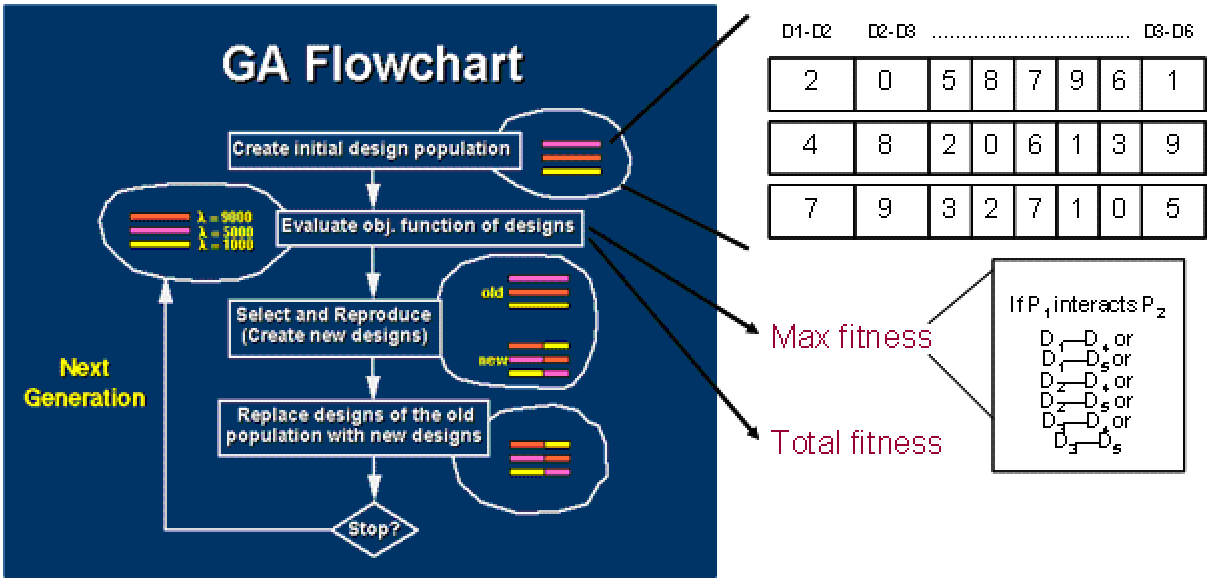

Obtain protein- domain information from Interpro
"Gold Standard" Data Set
Train
Test
Unique list of all domains
Domain Information for proteins
Unique list of all domains pairs (DDI)
Occurrence counts of DDI
Preprocessing
Subset selection
Prepare input files for GA
Genetic Algorithm
Implementation
Obtain Scores in "n" runs
Remove local maxima
Analysis of Results
Calculate mean, median DDI scores
Explanation ratios, precision, recall for PPI predictions

Figure 4.3: DomainGA methodology. Pre- and post-processing using python scripts forms important components of the entire methodology

4.2.1. Overview of the algorithm

Genetic Algorithms (GA) can be used as a search technique to find best-estimate solutions in optimization problems. They are a particular class of machine-learning algorithms that uses techniques inspired by evolutionary biology such as inheritance, mutation, recombination and natural selection. GAs are typically implemented as computer simulations in which a population of abstract representations (called chromosomes) of candidate solutions (called individuals) evolves toward better solutions. The solutions are either strings of 0/1s or can have different encodings. The evolution starts from a random population, and changes occur through a selection process over generations. In each generation, the fitness of the whole population is evaluated, and most successful individuals are kept for the next generation. This selected group of individuals is supplemented with offspring that are obtained by modifying (random mutations and/or recombinations obtained using crossovers with inherited characteristics) the individuals that are stochastically selected from the current population (with probabilities based on their fitness). The set formed this way then becomes the current population set in the next iteration of the algorithm.

In the DomainGA each chromosome is created as an array of DDI (parameter set to be optimized). It starts with 50 chromosomes that have randomly initialized parameter values as their array elements. An integer scale of [0-T] is used where T is 9 in all cases except in initial studies testing the dependence on the range of values as shown in Figure 4.4. In each generation the population size is increased 10-fold by the use of recombination, mutation, and random-generation operators. A multi-point recombination function is used among randomly selected chromosomes to add 250 (5x) more chromosomes. Random mutations are carried out on the genes of the chromosome (parameters) to create 150 (3x) new chromosomes. Finally, 50 (1x) random chromosomes are created and added to the initial population. When combined, this set forms the population of a particular generation. The chromosomes in the population are then evaluated based on an optimization fitness function, rank ordered and only the top 50 seed chromosomes are retained for the next iteration. The optimization fitness function is then maximized during the GA iterations, and when the score does not change over 15 successive iterations, the GA is terminated. At least 2000 GA runs starting from randomly selected populations are executed for each reported case. Finally, the results from the GA runs whose converged fitness function scores are lower than 80% of the maximum fitness value (local maxima) are considered unsuccessful, and are discarded from the statistical analysis that determines the distribution and mean values of the optimized parameter values reported.



D1-D2	D2-D3	....					D3-D6
2	0	5	8	7	9	6	1
4	8	2	0	6	1	3	9
7	9	3	2	7	1	0	5

Figure 4.4: Basic Genetic Algorithm operation. Shows a typical genetic algorithm flowchart with examples of the initial parameter vectors and the fitness functions tested in the DomainGA.

4.2.2. Optimization fitness function

The optimization fitness function evaluates how well the training PPI set is explained by the chromosome population. Each chromosome is an array of scores for the included domain-domain pairs, and this score set can be used to decide whether two proteins interact. Adapting the DDI scores to predict PPIs requires the development of a criterion to decide which type of domain-domain score corresponds to a PPI. For this, the DomainGA first forms a list of all possible DDI between two proteins; that is, all possible combinations between domain pairs. It then takes either the largest (maximum score detection rule) or the sum (total score detection rule) of the DDI scores from this list to represent the strength of the interaction between the proteins. If the determined strength is larger than a pre-assigned cutoff (>5 when T=9), then that pair of proteins is predicted to interact. The pair is assumed to not interact if the score is below the cutoff (<5 when T=9), and an indecisive assignment is made if it is equal to the cutoff. These PPI predictions are then compared to the training data where correct prediction is granted +1 point, and a penalty of −1 is applied for an incorrect identification. Indecisive assignments do not contribute to the optimization fitness score. As the number of positive and negative PPI entries in the training dataset can be vastly different, the contributions of the negative and positive PPIs are normalized to the overall fitness function according to the number of PPIs in each list such that both lists carry equal weight.

Since the selection of the negative set of PPI is a debatable issue with no best solution [47] an "alpha fitness function" is also provided in the DomainGA that uses only the positive PPI list as the training data and minimizes the magnitude of the involved parameters. This fitness function has two terms. The first term represents the explanation ratio of the training dataset; it is exactly the same function that was discussed in the previous paragraph. The second term is the sum of the magnitude squares of the parameters; that is, the sum of squares of all DDI scores. The second term is multiplied with a weight factor α and then subtracted from the first term to be used as the resulting fitness function in the GA runs. As discussed in the results section, results reported in Figure 4.7 were obtained using α=0.5, and the maximum score detection rule was used in deciding the PPI predictions. This fitness function maximizes the explanation ratio while assigning a minimum number of domain pairs as interacting partners. Note that without the second term, assignment of high values to all the optimized parameters would lead to a perfect explanation ratio of the positive PPI list so it would be a trivial global solution. The subtracted weighted parameter magnitude term blocks the optimizer from assigning high parameter values unless they are necessary to achieve a good explanation ratio.

4.2.3. Optimization parameter set selection

As in any multi-parameter optimization approach, the involved parameter set needs to be defined. For N domains there are N * (N+1)/2 possible domain-domain pairs whose values need to be known. Noting that N is on the order of 10^4 in InterPro classification, there are ~10^8 possible interacting domain-domain pairs. Reliable optimization of such large parameter sets requires PPI training data that are not currently available and possibly will not be available in the near future either. Therefore, inclusion of all possible domain-domain pairs in the optimization process is not realistic. To avoid parameter over-fitting, a small parameter set is selected and assumed to be large enough to represent the important domain, and thus protein, interactions. To select the used DDI parameter set, the histograms for the number of occurrences of the domain-domain pairs in the training PPI sets are created and the domain pairs are sorted according to their occurrence counts to achieve reasonably large training and test datasets. This procedure is repeated to select subsets with different number of domain-domain pairs (103, 867, 2466, and 5095 pairs; Table 1) to use as the parameter sets in the GA optimizations. Choice of these parameters was based on the occurrence of the domain pairs in the positive and negative standard PPI lists where roughly half of the parameters came from each list. It is important to note that DomainGA implicitly assumes that the omitted domain pairs are not a determining factor in deciding whether two proteins interact. Thus this would be equivalent to assigning zero (i.e., non-interacting) values to the neglected domain pairs.

4.2.4. Training subset selection

As discussed in Chapter 3 section 3.3, based on the parameter set selection, subsets of the gold standards training data are also created using the new concepts of *closed set* and *inclusive set*. It is important to note that in forming the closed dataset, an additional problem arises which reduces the number of parameters that can be truly optimized with the DomainGA method. Say that a domain pair is chosen as a parameter to be optimized. If all the PPIs that include this is domain pair d_{ij} contain at least another domain pair whose interaction score is not optimized (i.e., not a domain pair selected as a parameter), then these PPIs will be excluded from the list defining the closed set. This would lead to the case that domain pair d_{ij} may not appear in any of the PPIs defining the closed dataset. When that happens, as there is no information that is relevant for its optimization, the value of this parameter will be set randomly during the optimization. This was observed in the simulations and, when T was 9, such parameters had average values in the [4-5] range as expected. Thus, as they should, these parameters appear as fuzzy, uncertain parameters in the results. For this reason, whenever the closed and the inclusive set results are compared; these non-optimized parameters are omitted from the figures.

4.2.5. Assigning strengths to PPI

Domain GA method is set-up as a multi-parameter optimization method in which the extreme value of a fitness (score) function is searched. Adapting the DDI scores to predict PPIs requires the development of a criterion for deciding what domain-domain score corresponds to a PPI. For this, first a list of all possible DDI between two proteins is formed;

that is, all possible combinations between domain pairs. The largest or the total of the DDI scores is then taken to represent the ***strength*** of the interaction between the two proteins. If the determined strength is larger than a predetermined cutoff value, the protein pair is classified as interacting and as non-interacting otherwise. Throughout this chapter, the term *strength* is used in an unconventional manner. In this case, it is a score that represents the likelihood of interaction between two domains. The likelihood however has a bounded range and is discretized for practical implementation. Depending on the representation, the score can be interpreted as a normalized and scaled biochemical binding affinity or the thermodynamic Boltzmann factor, or as the statistical interaction probability. In the future versions of the algorithm, a continuous and unbound score range can be used which will make these correspondences more obvious.

4.3. Results and Discussion

There is no well established process of checking the correctness of a machine learning based prediction algorithm that depends so heavily on high quality training data. Therefore several tests are conducted to check for any assumptions introduced bias; statistical validation of PPI predictions based on cross-validation studies; cross-verification tests on test sets obtained from some other organism; discriminative & structural analysis of DDI score assignment; and finally testing the method by only using positive PPI for training. The following subsections discuss each of these tests in detail.

4.3.1. Testing the Robustness of DomainGA

Since the parameter set is reduced to avoid over-fitting the parameters during the optimization step, it raises the question of how dependent the derived values are on the size of the defined parameter set. A related concern is how representative the small set can be in terms of explaining the observations that are used as the training data. Another concern is the dependence of the optimization procedure on the score range used for parameter score assignment. Moreover optimization procedures such as Genetic Algorithms need to be tested for dependence on the fitness function or detection rule used. These issues have been addressed by the following test case studies to show that the selection procedure and these assumptions are reasonable.

4.3.1.1 Invariance with respect to the parameter score range: In the DomainGA, parameter values (i.e., the strength of each domain pair interaction) are optimized to maximize agreement with the training PPI list used. In the current implementation of DomainGA, the parameters are allowed to have integer values between 0 and T, where the upper bound determines the coarseness of the discretization. Figure 4.5 compares the results for the smallest data set when the maximum score value T was chosen as 5 and 9. The cutoff value to decide whether possible DDI result in a PPI was chosen as the mid-values 3 and 5 for the T=5 and 9 cases, respectively. Choosing the mid-values as the cutoff was totally arbitrary. In Fig. 4.5., the parameter scores are reported using a color scheme and the order of the parameters is the same in both parts. Each row in Fig. 4.5, shows the values of the parameter set (i.e., the domain interaction scores) optimized in a particular GA run. Each column shows the optimized value of a particular parameter across different GA runs. A uniform color through a column means that the corresponding parameter's score remain consistent across many different GA runs. Dominant red and blue colors represent interacting and non-interacting domain pairs, respectively, and other color shades correspond to intermediate parameter scores. The parameters with intermediate scores or whose values fluctuate between high and low scores across the different GA runs as defined as fuzzy (or indefinite) parameters. It is clear from Fig. 4.5. that the scale choice does not make a noticeable difference. A correlation analysis of the optimized parameter values computed as the mean of the GA runs indicates an almost perfect match with an R-square value of 0.9996 between the T=5 and 9 cases.

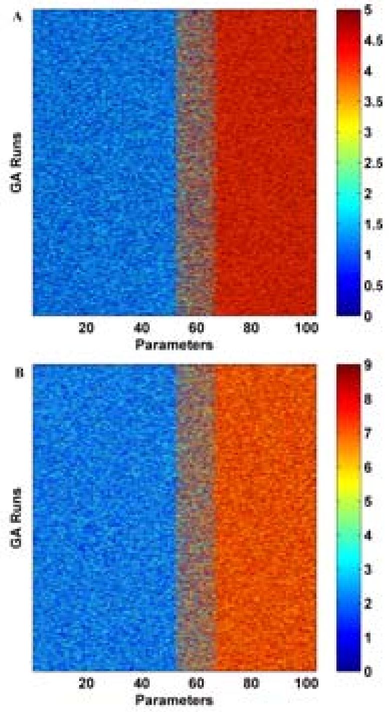


Figure 4.5: Comparison of the scores of the common 103 parameters. These were optimized using different ranges for the scores with the inclusive set. Employed range was: (A) [0-5] and (B) [0-9]. In the figures, the vertical axis represents a particular GA run and the horizontal axis shows the optimization parameters, which are rank ordered according to their mean strength values. Each column shows the score of a particular parameter obtained in different GA runs. A consistent color through a column indicates that the optimized value

of corresponding parameter is almost the same in all the GA runs. Each plot reports the optimized score set values for more than 2,000 GA runs. Intense blue and red colors respectively represent the non-interacting and interacting domain-domain pairs.

4.3.1.2. Invariance with respect to the number of parameters: In an optimization study, an added concern is the dependence on the size of the parameter set. To address this issue, datasets have been created with different number of parameters, Table 4.1. As discussed in the Research Methodology section, dataset with 867 parameters was selected based on single- and multiple-occurrence statistics of the domain pairs in the training set. The size of this dataset was further increased to 2466 and then to 5095 by adding more parameters to the list (Table 1). It should be noted that the parameters of the 867-parameter set are a subset of the larger parameter sets. Inclusion of the same parameters in several datasets allows numerical tests to see if the optimized values of the parameters depend on the size of the set used. Figure 4.6 reports the optimized values for the 867 parameters that are common in all sets. Comparison of the results shows that the assignments of a small fraction (~15%) of the parameters change between the high, low, or fuzzy categories. Therefore, vast majority of the domain-domain pair interaction scores do not depend on the included number of optimization parameters. The most noticeable pattern between the results for the cases is that, as the number of optimized parameters is increased, scores of some of the parameters shift from the positively determined to the fuzzy (indeterminate) category, Fig. 4.6D. The differences however do not alter the explanation ratios of the training datasets, Table 4.2.

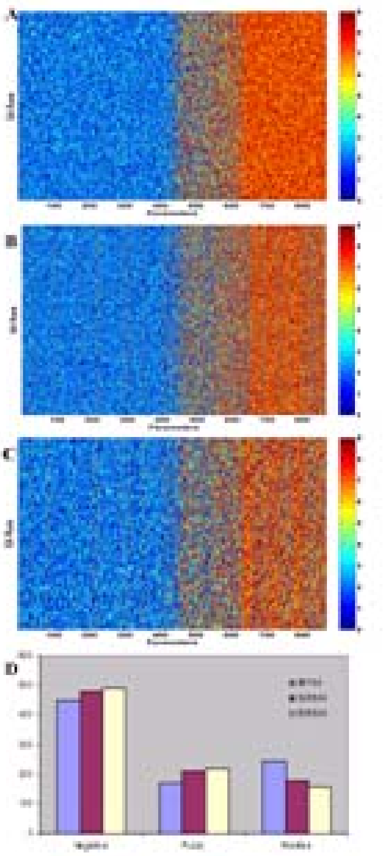


Figure 4.6: Comparison of scores between the optimization studies with different number of parameters. Parts (A-C) report the scores of the 867 parameters that were common in all three cases. Inclusive set optimizations with: (A) 867; (B) 2466; and (C) 5095 parameters. Part (D) reports and compares the classification of the optimized scores according to their interaction profiles.

4.3.1.3. Invariance with respect to the detection rule: The robustness of the DomainGA method with respect to the detection rule choice is tested by comparing results obtained using the *max-score detection rule* and the *total-score detection rule*. In terms of biophysical considerations, the maximum-score detection rule emphasizes the dominant DDI, and it implicitly assumes that proteins interact through, at most, one domain at a time, and the domain-domain pair with the highest affinity is the most crucial one. In contrast, the total-score detection rule considers all possibilities by summing over the interaction score, which is analogous to calculating the cumulative thermodynamic free energy of a PPI where every possibility contributes according to its strength.

Optimizations using both of the detection rules were carried out using the closed 344 parameter set (Table 4.1). The parameter score range was [0-9] and a cutoff of 5 was used to classify the PPIs into positive or negative interaction categories. Parameter values obtained using the total- and the maximum-score detection rules are compared in Figure 4.7. As the reported two-dimensional histogram shows, the scores of the domain pairs in these two optimization studies lie close to the diagonal demonstrating the promise that the DomainGA results are rather insensitive to the detection rule. There are only a few parameters that have conflicting optimized values between the two detection rule cases. These appear as a spike at the (max~7, total~1) point in the histogram diagram indicating a discrepancy between the parameter sets. It is important to note that the small differences at the low or high parameter scores are unimportant because in the current classification scheme, values are simply grouped into three classes: non-interacting (<5), fuzzy (~5), and interacting (>5). Therefore, small variations in the (0:3) or (7:9) ranges are irrelevant to the derived conclusions.

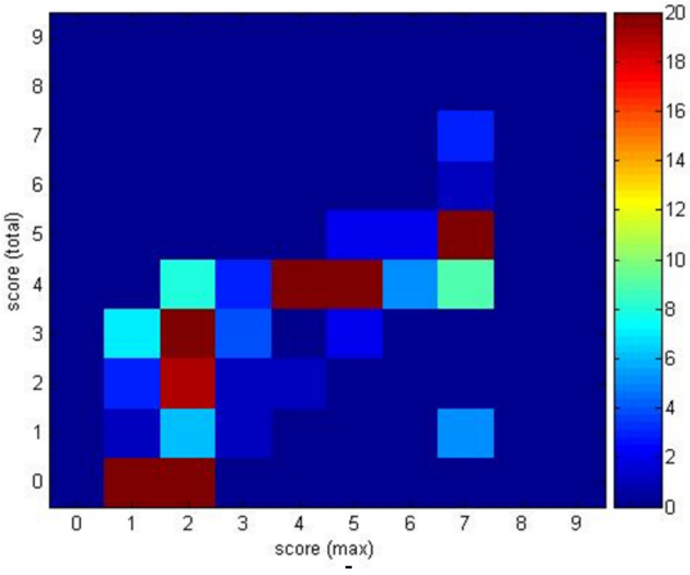


Figure 4.7: Comparison of the optimized parameter scores. There were optimized using the 344 parameter closed set with maximum (x-axis) and total (y-axis) score detection rules. Reported scores are the averages of the GA runs after the infrequently occurring parameter values are discarded during analysis. Histogram diagram reports the score distribution of the parameters that can be optimized in the simulations. Each (x,y) entry in this histogram plot reports the number of parameters that has mean values of x and y when the maximum- and total-score detection rule was used in the optimization, respectively. The maximum value of the color scale is lowered from 67 to 20 to enhance the contrast between the histogram points.

4.3.1.4. Comparison of subsets of training data: To test the robustness of the algorithm with respect to the sub-setting of the training data (closed and inclusive sets) comparison of the mean scores for the common 344 parameters (DDI) in both the sets was conducted. The results of the two optimization studies agree very well, which is evident in Figure 4.8.

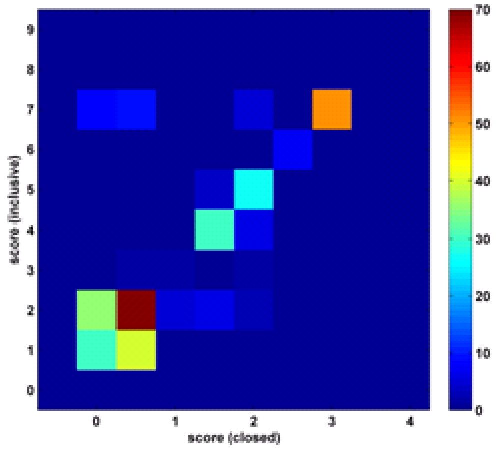


Figure 4.8: Comparison of scores of the 344 common parameters. There parameters are common between the closed 344 parameter (x-axis) and inclusive 867 parameter (y-axis) datasets. The maximum score detection rule was used and the reported scores are the averages of the GA runs after the infrequently occurring parameter values are discarded during analysis. Each (x,y) entry in this histogram plot reports the number of parameters that has mean values of x and y when the referred closed and inclusive dataset was used in the optimization, respectively.

4.3.1.5. Parameter space search: One major concern in a parameter optimization study is the appropriate sampling of the parameter space. In the GA runs, initial values of the parameters were picked randomly and the optimized parameter values were statistically analyzed. Results reported in Figures 4.5 & 4.6 are representative of our typical findings. In these figures, each row shows the optimized values of the parameter set in a particular GA run, and a uniform shade across a column means that particular parameter has the same optimized value at the end of every GA run. It is clear that optimal solutions of the GA runs have insignificant variations in the optimized parameters when the score of a parameter is in the low or high categories; that is, if the parameter indicates that a domain-domain pair is found to be interacting or not. DDI parameters that are in the fuzzy range, i.e., may or may not interact, generally have larger variations. This is to be expected because these fuzzy parameters are indefinite and do not contribute much to the information content of the machine-learning step. Thus, the overall explanation ratios of the training set are rather insensitive to their variations.

4.3.2. Cross-Validation studies

The previous sections have discussed the robustness of the DomainGA method to the selection of the parameter sets, score ranges, detection rules, and search space. The ***explanation ratio*** of the training dataset can be one evaluation criteria to determine the success of a machine-learning method. The explanation ratio is defined as the percentage of the PPIs in the training set that are successfully accounted for at the end of an optimization, i.e., it is the ratio of correctly predicted to the total number of entries in the list. Let TP, TN, FP, and FN respectively stand for true- and false-positive and negative predictions. Then, the explanation ratios of the training sets are TP/(TP+FN) and TN/(TN+FP) for the positive and negative PPI lists, respectively.

Another way to evaluate the performance of the optimizations is to conduct a cross-validation analysis with a testing dataset. *N*-fold cross validation with *N*~10 is typical in machine learning studies where ~10% of the entries are used for testing the predictions based on training with the remaining 90%. This process is repeated 10 times for the data split 10 ways. Performed 10-fold cross validation test using the inclusive 867 parameter dataset indicated that the DomainGA optimization achieves an average explanation ratio of 94.8% and 97.0% for the training and 92.9% and 97.0% for the testing sets for the positive and negative PPIs, respectively. A two-fold test was also conducted in which half of the dataset was used as the training data, while the remainder served as the testing dataset. In this most severe form of *N*-fold cross validation, DomainGA optimization achieves an average explanation ratio of 95.7% and 96.6% for the training and 88.8% and 96.5% for the testing sets for the positive and negative PPIs, respectively. Overall, these are very respectable results for *N*-fold cross validation tests.

No of Parameters	**Training set**	**PPI**	**Explanation Ratio (%)** [a]	**Accuracy**	**Precision**
867	Inclusive	Positive	95.6	96.4	38.9
		Negative	96.1		
344	Closed	Positive	99.3	98.7	90.6
		Negative	98.7		
2466	Inclusive	Positive	96.2	95.7	24.3
		Negative	95.7		
1216	Closed	Positive	99.0	96.6	60.8
		Negative	96.4		
5095	Inclusive	Positive	97.3	95.6	19.3
		Negative	95.6		
3060	Closed	Positive	99.3	95.9	56.5
		Negative	95.6		
Deng et al.	Inclusive	Positive	95-98	93-95	24-30

with 867 pmts [b]		Negative	93-95		
	Closed	Positive	91-93	89-90	54-55
		Negative	89-90		
Random with 867 pmts	Inclusive	Positive	61.0	36.0	2.2
		Negative	35.4		

Table 4.1. Explanation ratios of the MIPS yeast datasets

[a] Explanation ratio is the ratio of successful predictions to the overall number of entries in a particular list. Note that the explanation ratios for the positive and negative PPI lists respectively correspond to the sensitivity and specificity with Lin et al. definitions.

[b] Calculated performance metrics depend on the false positive and negative prediction rates used in the MLE algorithm as well as on the cutoff for positive and negative PPI assignments. Therefore, the reported range of percentages were obtained when various prediction rates were used in the MLE algorithm.

4.3.3. Comparison with existing approaches

The information contained in the calculated explanation ratios relates to the content of the Receiver Operating Characteristic (ROC) curves that are often reported in machine-learning studies [32, 69, 70]. DomainGA approach achieves explanation ratios that are larger than 95% for the parameter sets used (Table 4.2). The performance of the DomainGA method is evaluated using the sensitivity=TP/(TP+FN) and specificity=TN/(TN+FP) definitions given by Lin et al. [69]. Note that these properties are equal to the explanation ratios for the positive and negative PPI lists, respectively, that are reported in Table 4.2. Thus, these optimizations typically result in predictions with a >95% sensitivity and >95% specificity, which is equivalent to a point (0.05, 0.95) in the ROC plot indicating a very steep curve, a highly desired attribute.

Martin et al. developed their own set of definitions for performance evaluation [38]. They define the additional benchmark measures of accuracy = (TP+TN)/(TP+FP+TN+FN) and precision= TP/(TP+FP). Obtained values for these measures are reported in Table 4.2. Accuracy and precision of the DomainGA predictions with the inclusive 867 parameter set are 2.7 and 17.7 times higher than the random predictions, respectively. Having a much better precision with the closed set compared to the inclusive set is most likely due to the implicit assumption that the excluded parameters do not contribute to the predictions. This assumption is not needed in the closed set studies but it can be severe for certain protein pairs and may limit the precision of the predictions in the inclusive set cases. Therefore, as the representation is contained in itself, even though the number of truly optimized parameters is less (Table 4.1), optimization with the closed datasets can achieve a much higher precision. Another trend that is obvious in these results is that the precision decreases with the increase in the number of included parameters. As discussed above, this is most likely due to the limitation with the amount of information to reliably optimize some of the parameters included in the larger 2466 and 5095 parameter sets.

The predictions of the DomainGA method can be compared with the results obtained by the Maximum Likelihood Estimation (MLE) method of Deng et al. [22]. The MLE method was re-implemented and various false positive and negative prediction rates were experimented since they are necessary parameters during the likelihood maximization stage. The results depict that the overall results are rather insensitive to the used false negative and positive prediction rates. The same conclusion was also reached by Deng et al. themselves

[22]. The results for the MLE prediction are reported in *Table 2* for various case scenarios. Explanation ratios (which also correspond to the sensitivity and the specificity) achieved by the MLE method are slightly lower than our DomainGA predictions. The accuracy obtained by the MLE is 90% for the closed and 94% for the inclusive datasets, which are lower than the accuracy of the DomainGA results, 96% and 99%. However, the most notable difference is in the precision of the predictions. Even though the precision of the DomainGA may appear to be low, 91% for the closed and 39% for the inclusive sets, it is considerable higher than the precision of the MLE method, 55% for the closed and 30% for the inclusive sets. It should be noted that both methods perform much better than the random predictions.

4.3.4. Cross verification studies

Although cross validation in machine learning studies is important, when the training and testing data are of the same origin, this may bias the predictive power of a method. For this reason, the cross-explanation ratios for the DomainGA optimization results are also computed (Tables 4.3 & 4.4), which aids in verifying the results across datasets of different origin. In the cross verification tests, the parameter sets were optimized using one set of training data (MIPS yeast data in this case) and then the predictive power of the optimized parameter set is checked by computing the explanation ratio of another set (e.g., another yeast dataset or the human PPI data) that has not been used during the training. This is analogous to an extreme form of cross-validation because training and testing sets may not have much resemblance; therefore, an algorithm passing this type of testing would show its wider predictive power and applicability. This argument is also valid for using the closed and inclusive set combination from the same resource for training and testing purposes, albeit to a lesser degree.

Analysis of the yeast results shows that when MIPS datasets are used for training, DomainGA optimization can achieve remarkable explanation ratios of the training datasets, typically at higher than 95% level (Table 4.2). Since all of the domain pairs that appear in the used training set are included as parameters in the optimization, as expected, explanation ratios are slightly higher for the closed set cases. Using the optimized parameter values, the cross-explanation percentages between the MIPS yeast datasets are computed. These calculations (Table 4.3) showed that parameters optimized using the inclusive set explains the closed set data extremely well – typically at the 99% and 96% level for the positive and negative PPIs, respectively. These ratios are nearly as good as the ratios obtained by training on the closed set itself (Table 4.2). This may be expected because, as they are a subset of the inclusive set, the closed set data are included in the computations. On the contrary, the parameters optimized using the much more limited closed set are less successful in explaining the inclusive datasets (Table 4.3); however, its success is still quite respectable. Since the closed set starts to represent the inclusive set better, the cross explanation ratios improve with the increase in the size of the parameter set, Table 4.3. As a further check, comparison of the optimized parameter scores shows that the use of the closed and inclusive datasets results in very similar parameter values (Figure 4.5). The parameters whose optimized values disagree between the methods appear as off-diagonal elements in the lower right or upper left corners in Figure 4.5; clearly, only a very small percentage of the parameters exhibit this behavior.

Not surprisingly, cross-verification studies between the MIPS and Uetz et al. yeast datasets resulted in lower explanation ratios (Table 4.3). However, the explanation ratios are still at a very respectable ~75% level. Information about the PPI networks in yeast collected in various high-throughput studies is known to have small overlap [65]. This is expected to be reflected in this scheme as well where the domain-domain pairs that are selected to represent the MIPS datasets may not contain the necessary decisive information that represent the Uetz et al. datasets [4, 64]. Corollary to this would be that the domain-pairs that are important to represent the Uetz et al. data were not included in our optimization studies because, based on their occurrence, they were not among the most important ones in representing the MIPS PPI dataset.

Training set	**Test set**		**Explanation Ratio (%)** [a]
867 pmt inclusive	344 pmt closed	Positive	99.3
		Negative	95.4
344 pmt closed	867 pmt inclusive	Positive	69.9
		Negative	64.3
2466 pmt inclusive	1216 pmt closed	Positive	98.6
		Negative	96.0
1216 pmt closed	2466 pmt inclusive	Positive	76.7
		Negative	66.5
5095 pmt inclusive	3060 pmt closed	Positive	99.3
		Negative	95.6
3060 pmt closed	5095 pmt inclusive	Positive	84.5
		Negative	67.8

867 pmt inclusive	Uetz et al. [b]	Core	78
		Full	75
344 pmt closed	Uetz et al. [b]	Core	78
		Full	75

Table 4.2. Cross verification with the yeast datasets

[a] Explanation ratios were calculated by using the indicated yeast datasets as the testing data after the DDI score parameters were optimized using the reported MIPS set as the training data in the DomainGA runs.

[b] Explanation ratios were calculated by using the indicated closed Uetz et al. yeast datasets [4] as testing data. Uetz et al. datasets (http://dip.doe-mbi.ucla.edu/dip/ Download.cgi) contain only positive PPIs so test statistics were computed only for the positive predictions. Yeast Core set originally contained 5952 positive PPIs of which 74 were retained after selecting the entries according to their relevance to the parameter set utilized in optimizations. In the Yeast Full dataset case the corresponding total and retained values were 17471 and 119.

In another cross-verification study, the DDI scores that were optimized using the MIPS yeast data were used to compute the explanation ratios for the human interactome (Table 4.4).

MIPS Training set	**Closed HPRD Test set**	**Explanation Ratio (%)**
344 pmt closed	Positive	75.4
	Negative	92.9
867 pmt inclusive	Positive	75.5
	Negative	93.7
Random scores	Positive	70.0
	Negative	35.9

Table 4.3. Cross verification with the human PPI *

* Explanation ratios were calculated by using the indicated closed human PPI [40] datasets as testing data after the DDI score parameters were optimized using the reported MIPS set as the training data in the DomainGA runs. Explanation ratio is simply the ratio of successful predictions to the overall number of entries in the used positive or negative PPI list. Human PPI dataset was obtained from the Human Protein Reference Database (HPRD) and it has contained 364645 positive and ~40 million negative PPIs. Of these 215 and 6892 were retained respectively after selecting the entries according to their relevance to the parameter set utilized in optimizations.

Explanation ratios obtained for the closed sets were 74% and 93% for the positive and negative PPI sets, respectively. These are surprisingly high percentages, particularly for the negative protein interaction predictions. Explanation ratios obtained with the DomainGA method can also be compared to the predictions of a random score scheme. As Table 4.4. indicates, the DomainGA method significantly improves on the random predictions, particularly for predicting the non-interacting protein pairs. Accuracy (93%) and precision (28%) of the DomainGA is much higher than the corresponding values for the random predictions with 37% accuracy and 3.3% precision. Thus, the DomainGA increases the precision of the across-organism predictions by a factor of 8.4 and, based on the severe cross-verification test, it can be concluded that the DomainGA method shows great promise to be applicable across multiple organisms.

4.3.5. Evaluation of the obtained DDI scores

As discussed in the Introduction section, a rationale behind the presented research was the lack of discriminatory power of the InterDom DDI scores. To further evaluate the DomainGA method's performance, a similar analysis has been conducted using the DomainGA interaction scores. Figure 4.9 reports the distributions of the predicted yeast PPI scores obtained using the DDI scores obtained in the inclusive 867 parameter study. Using the same optimized parameter values, as in the cross-verification study reported above, Figure 4.9 also reports the predicted score distribution for the human interactome for the closed PPI dataset. For both cases, distributions for the positive and negative PPI scores are clearly well separated indicating that, in terms of having discriminatory power, the DomainGA method significantly improves on the InterDom scores.

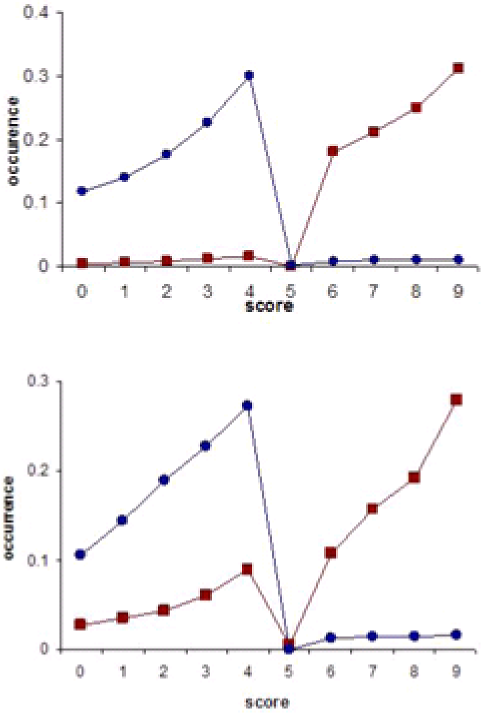


Figure 4.9: Comparison of the strengths of the MIPS positive and negative PPIs. MIPS positive (red line with squares) and negative (blue line with circles) PPIs are computed using the DomainGA optimized DDI scores. Vertical axis shows the percentage of the PPIs with interaction scores that were calculated by binning the total PPI scores using unit bin sizes. Top: Inclusive set yeast PPI; Bottom: Closed set human PPI.

4.3.6. Testing against structurally identified interactions

The iPfam resource [71] makes use of the biomolecular structures deposited in the protein data bank (PDB) and identifies the possible interactions between the domains defined by the Pfam classification. Because it is based on structural information, derived DDI can be considered reliable. However, it should be kept in mind that iPfam uses an automated computational approach and does not distinguish between biological and crystal contacts. In addition, interactions between the domains of a single polypeptide and domain interactions between separate peptides are not treated separately. These characteristics can lead to false-positive detections in iPfam. The domain interactions between Pfam-A class domains from the iPfam website were used to investigate the corresponding scores that were obtained in the DomainGA studies. As pointed out by Deng et al. [22] due to the unavailability of high number of known DDI, it is difficult to estimate the accuracies of interaction predictions at the domain level. As an evidence of this, out of the 867 domain-domain pairs included in the most precise optimization study only 33 were found to be included in the iPfam list (Table 4.5).

Domain Name (Pfam ID)	**Domain Name (Pfam ID)**	**Mean Score [a] (Inclusive/Closed set)**
PNPase (PF03726)	RNase_PH (PF01138)	7.61/7.48
GTP_EFTU (PF00009)	GTP_EFTU (PF00009)	7.57
Ribosomal_L6 (PF00347)	Ribosomal_L6 (PF00347)	7.56
CK_II_beta (PF01214)	CK_II_beta (PF01214)	7.56/7.49
Prenyltrans (PF00432)	PPTA (PF01239)	7.53
Ribosomal_S8 (PF00410)	Ribosomal_S2 (PF00318)	7.52
TPR_1 (PF00515)	TPR_1 (PF00515)	7.52
Ribosomal_S11 (PF00411)	Ribosomal_S7e (PF01251)	7.52/7.50
IF-2B (PF01008)	IF-2B (PF01008)	7.51/7.49
CK_II_beta (PF01214)	Pkinase (PF00069)	7.49
Ribosomal_S2 (PF00318)	Ribosomal_S2 (PF00318)	7.49
Bromodomain (PF00439)	Bromodomain (PF00439)	7.48/5.66

WD40 (PF00400)	G-gamma (PF00631)	7.48/1.97*
Ribosomal_L4 (PF00573)	Ribosomal_L37e (PF01907)	7.48
G-alpha (PF00503)	WD40 (PF00400)	7.46
PFK (PF00365)	PFK (PF00365)	7.45
Ribosomal_S8e (PF01201)	Ribosomal_S2 (PF00318)	7.44
GTP_EFTU (PF00009)	EF1_GNE (PF00736)	7.43
Proteasome (PF00227)	Proteasome (PF00227)	6.26/6.11
ATP-synt_ab (PF00006)	ATP-synt_C (PF00137)	5.86
Clat_adaptor_s (PF01217)	Adaptin_N (PF01602)	5.40
Ribosomal_L4 (PF00573)	Ribosomal_L15e (PF00827)	5.38
Glyco_transf_20 (PF00982)	Glyco_transf_20 (PF00982)	5.33
Ribosomal_L24e (PF01246)	Ribosomal_L14e (PF01929)	5.30/5.39
Prefoldin (PF02996)	KE2 (PF01920)	4.91
Proteasome (PF00227)	AAA (PF00004)	4.75/4.92
Pkinase (PF00069)	Pkinase (PF00069)	2.43*
WD40 (PF00400)	WD40 (PF00400)	2.11/2.04*
RRM_1 (PF00076)	RRM_1 (PF00076)	2.07*
Pkinase (PF00069)	Ank (PF00023)	2.05*
Myb_DNA_binding (PF00249)	Myb_DNA_binding (PF00249)	2.04*
WD40 (PF00400)	PH (PF00169)	1.92/2.04*
Ank (PF00023)	Ank (PF00023)	1.87*

Table 4.4. DDI scores for the pairs that appear in the iPfam database*

* Entries are discussed in the main text.

a Scores obtained in the inclusive 867 parameter study. In the entries that contain more than one score, the second numbers are the scores for the domain pairs that were obtained in the corresponding 344 parameter closed set optimizations.

According to the score predictions with the inclusive set, seven of these domain pairs have low scores thus reflecting a disagreement between these results and the information listed at the iPfam database. Detailed investigation of these eight conflicting cases conducted by Dr. Haluk Resat at PNNL was illuminating for evaluating the success of the DomainGA method as discussed in Singhal et al. [72]. This demonstrates that in addition to helping with constructing PPI networks, this domain-based approach may also be of use in detecting the biophysical properties of the protein functional domains.

4.3.7. Optimization using only the positive PPIs

Although the positive PPI lists are generally based on direct experimental observation, the negative PPIs can be ambiguous as discussed by Ben-Hur et al. [47]. As in the compilation of the MIPS dataset that is used, negative interactions are often extracted by making certain assumptions; for example, proteins that occupy different sub-cellular compartments do not interact. Implicit in this assumption is that the proteins would still not interact even if the biophysical barriers keeping them in separate compartments are removed. This in essence is a severe assumption whose correctness is questionable, and the assignment of locations can itself be problematic [73]. To test the utility of DomainGA without any negative PPI dataset, optimizations have been conducted with a different GA optimization fitness function that maximizes the explanation ratio of the training dataset while keeping the values of the domain-domain score parameters at a minimum. The idea of minimizing the number of positive domain interactions is analogous to choosing a smaller set of domain-pairs with higher-specificity concept that was advocated in reference [74].

Comparison of the results obtained using only the positive MIPS PPI dataset for the closed 344 parameter case with the new minimum parameter magnitude fitness function (details of the optimization routine are described in the Research Design and Methodology section) with the above reported results shows very good correlation between the results (Figure 4.10). In line with the earlier cases, the explanation ratio of the training set was very high (98%). To test whether the unused negative PPI list was still well predicted with the obtained scores, the explanation ratios were computed, and it was 96%, an excellent ratio. Thus, with the use of realistic fitness functions in the GA optimization runs, one may be able to sidestep the problems associated with the availability of the negative PPI training data.

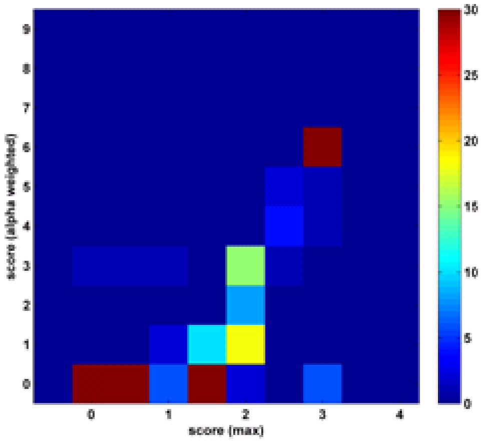


Figure 4.10: Comparison of the mean scores of the parameters optimized using the 344 parameter closed set training data with different fitness functions. X-axis: Optimization using both the negative and positive PPIs with the maximum score detection rule;. Y-axis: Optimization with the minimum parameter magnitude fitness function using only the positive PPI list. The maximum value of the color scale is lowered from 121 to 30 to enhance the contrast between the histogram points.

One clear trend in the optimized values of the parameters is that scores for the domain-domain parameters are generally lower with the new optimization fitness function (Figure

4.10). This is an expected outcome because, as a result of the way the optimization fitness score is constructed, the algorithm would only set a minimal number of parameters to have large non-zero values. The shift in the values of the scores does not create any discrepancy between the results. There are 44 parameters that have values >5 in the optimization with the new fitness function. This finding for the number of interacting domain pairs is in accord with the predictions of the closed set optimization runs that use both the positive and negative PPI lists as training data.

CHAPTER FIVE
DOMAINSVM

Abstract

DomainGA methodology introduced in Chapter 4 utilized the structural domain information in the PPI to develop a scoring scheme for domain-domain interactions. Although it accounted for “OR” combinations of domain interactions in the Maximum-Fitness function and the “AND” combinations of domain interactions in the Total-Fitness function; it did not account for “AND/OR” combinations of multiple domain interactions. The chapter presents the DomainSVM method which is a support vector machine based approach that classifies protein-protein interactions by accounting for combinations of domain-domain interaction scores obtained from DomainGA. DomainSVM method outperforms PPI predictions obtained from DomainGA by achieving very high explanation ratios, precision, specificity, sensitivity and F-measure values in a 10 fold cross-validation study conducted on the positive and negative PPIs in yeast. Functional analysis of the predicted interactions amongst unknown protein pairs brings forth interesting observations.

5.1. Introduction

As discussed in the Chapter 2, most of the existing domain based approaches for protein interaction prediction assume independence of domain-domain interactions. As pointed out by Dohkan et al. [33] and Han et al. [26], the multiple complex structures in the Protein Data Bank (PDB) suggest that multiple domains take part in a physical interaction. Therefore it is essential to consider the effect of multiple domains interactions when predicting protein-protein interactions.

This chapter introduces the DomainSVM approach which uses the interaction information within domains of proteins to infer biological PPI networks. DomainSVM is a two step process in which first the interactions between the functional domains of the proteins are quantified using the DomainGA method and then those scores are used in an SVM based technique to detect patterns of domain interactions for classifying protein-protein interactions. Similar to the DomainGA approach this algorithm is benchmarked using the PPI data available for *S. cerevisiae*. The following sections discuss the two steps of this technique (1) optimization of the domain-domain interaction scores and (2) use of the optimized scores in an SVM setup to predict PPI interactions. The results are then presented which show the statistical validity of the approach using measures such as accuracy, precision, recall and F-measure values on a 10-fold cross validation study conducted on positive and negative PPI for yeast. The PPI predictions obtained from DomainSVM are then compared with predictions from DomainGA on the same datasets using statistical measures such as sensitivity and specificity; and the incorrect predictions of DomainGA correctly predicted in DomainSVM are analyzed. In addition interesting observations obtained from biological function analysis of the different categories (true positives, false positives, true negatives, false negatives) of predicted PPI are presented.

5.2. Research Design and Methodology

Figure 5.1 depicts the flow of logic in the DomainSVM methodology which is discussed in detail in this section.

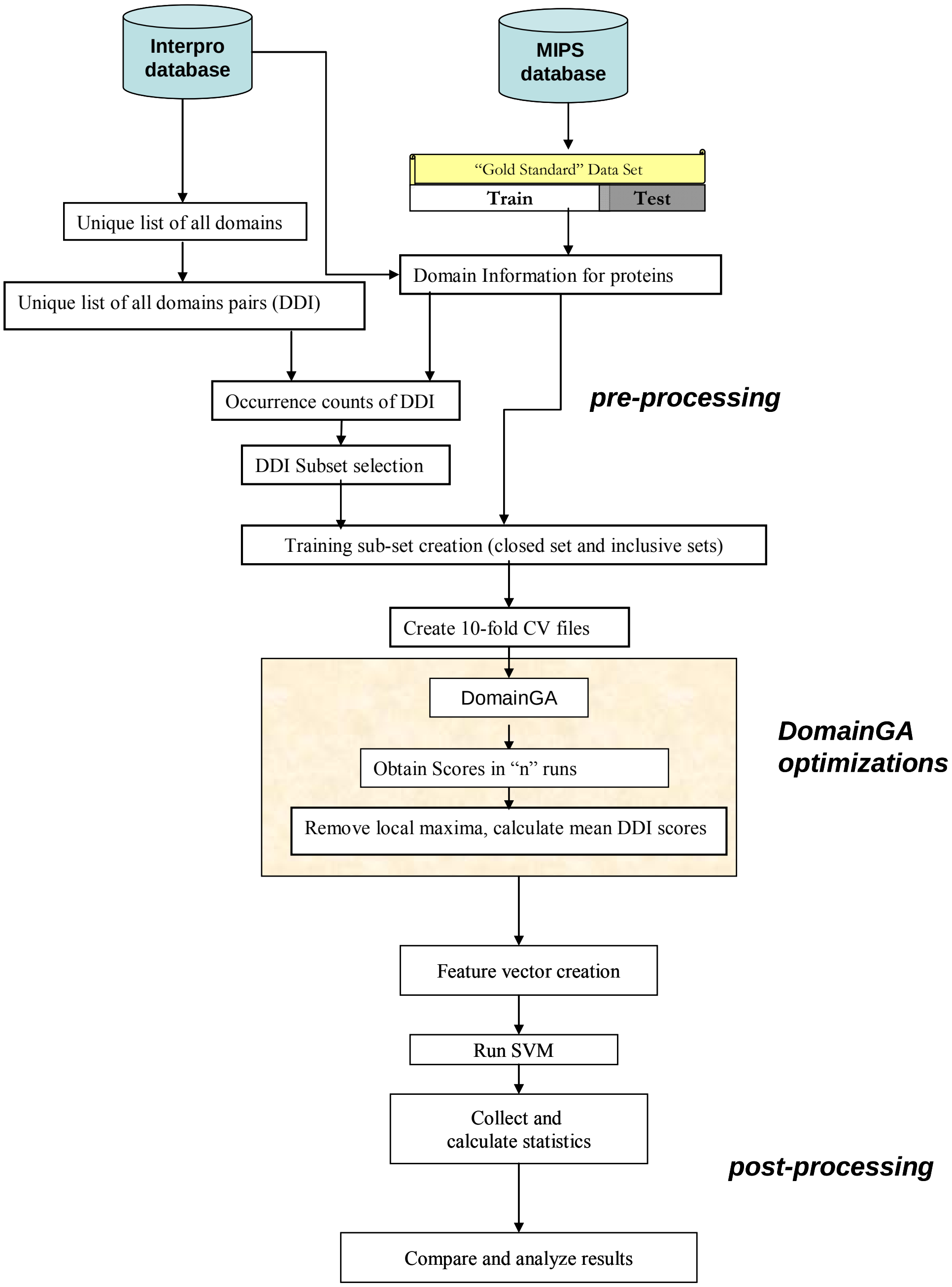

Interpro database
MIPS database
"Gold Standard" Data Set
Train
Test
Unique list of all domains
Domain Information for proteins
Unique list of all domains pairs (DDI)
Occurrence counts of DDI
pre-processing
DDI Subset selection
Training sub-set creation (closed set and inclusive sets)
Create 10-fold CV files
DomainGA
DomainGA optimizations
Obtain Scores in "n" runs
Remove local maxima, calculate mean DDI scores
Feature vector creation
Run SVM
Collect and calculate statistics
post-processing
Compare and analyze results

Figure 5.1: DomainSVM methodology. Pre- and post-processing using Python scripts forms important components of the entire methodology.

5.2.1. Overview of the approach

DomainSVM makes use of the DDI scores optimized using DomainGA in an SVM based learning approach to detect multiple domain pairs working together to predict PPI. As discussed in Chapter 3 and depicted in Figure 5.1, the gold standards PPI data is obtained from the MIPS database. The data is validated to remove duplicates and other inconsistencies such as interactions appearing in both positive and negative samples. The domain information for the PPI in the GSTD list is obtained from the Interpro database and a list of possible domain pairs is formed along with occurrence statistics of DDI in GSTD PPI. Two subsets of the DDI are created with 867 and 5095 elements each based on high occurrence characteristics. Based on the two subsets of the domain-pairs, four subsets of the GSTD PPI data are created namely: 867-closed set, 867-inclusive set, 5095-closed set and 5095-inclusive set. These four sets are then subset into 10 sets each to conduct 10-fold cross-validation (CV) studies keeping 1/10th (X) of the positives examples and randomly selected(without replacement) 4X negative examples from the parent list in each of the 10 CV sets. Once the 10 fold CV sets are created for each of the 4 sets being studied, DomainGA optimizations were conducted for each of the 40 sets. The setup of an optimization in DomainGA requires providing a list of parameters (DDI) to be optimized, a range for score assignment, a set of gold standard PPI and the selection of the fitness function to be used. The maximum score detection rule in which the DomainGA uses the largest of the domain-domain interaction scores for each protein pair to represent the strength of the interaction between the proteins was used. The optimization fitness function is maximized during the GA iterations, and when the score does not change over 15 successive iterations, the GA is terminated. At least 2000 GA runs starting from randomly selected populations are executed for each reported case. Finally, the results from the GA runs whose converged fitness function scores are lower than 80% of the maximum fitness value (local maxima) are considered unsuccessful, and are discarded from the statistical analysis that determines the distribution and mean values of the optimized parameter values used in the SVM as discussed in section 5.2.2. Finally the SVM runs were conducted using SVMLight [75] implementation as discussed in section 5.2.3 and several statistics were gathered as discussed in the Results and Discussion section 5.3. All these steps were repeated for each of the 40 sets created. Since most of these tasks need to be performed regularly on different subsets of the data they were automated by developing Python scripts for the same.

5.2.2. Utilizing domain-domain interaction scores in an SVM

To consider the effect of combinations of domains interactions on protein-protein interactions prediction, DomainSVM uses the domain-domain interactions scores obtained from DomainGA in a Support Vector Machine based classifier. Support Vector Machines were developed by Vapnik [76, 77] for binary classification and regression estimation tasks. Since their introduction, SVMs have been used in a large number of bioinformatics tasks such as homology detection [78-81], sub-cellular localization prediction [82-84] and protein interaction prediction [18, 33] among others. SVMs are machine learning algorithms designed with the intention of “generalizing better”. The problem SVM algorithms address relates to the efficient learning of a classification rule from a set of exemplars.

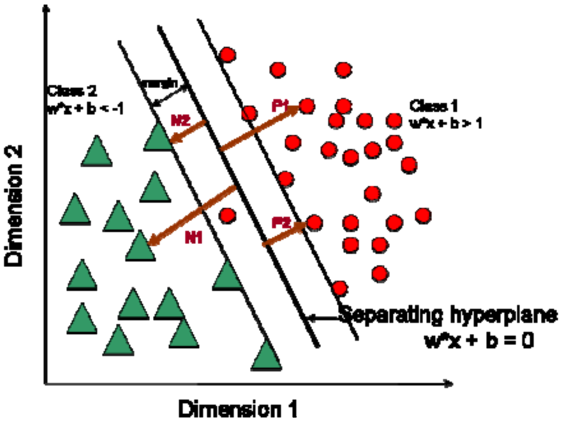


Figure 5.2: Working of a Support Vector Machine. This figure shows the representation of positive and negative examples in two dimensional space separated by the hyper-plane.

As shown in Figure 5.2, in classification tasks, the goal of the SVM is to determine whether a candidate “belongs to” or “does not belong to” a given class. A linear SVM has a decision function of the form:

$$f(x) = wx + b \quad (1)$$

where w is the weight vector and b is a constant bias. A data point is classified according to the sign of the function f. The choice of w and b is such that the separation between the positive and negative examples is the maximum for linearly separable data points. One requirement of SVM algorithms is that the input be represented as a set of fixed-length vectors. The internal transformation of the input fixed-length data vectors into a non-linear high-dimensional feature space can be accomplished by means of a kernel function. Any symmetric, positive semi-definite function is a valid kernel function, corresponding to an inner product in some feature space. The base kernel in an SVM is generally normalized forcing each vector to have a length of 1 in the feature space i.e.

$$K(X,Y) = \frac{X \cdot Y}{\sqrt{(X \cdot X)(Y \cdot Y)}} \qquad (2)$$

The existing methods [33] using domain information in an SVM for predicting PPI either use the domains in one protein concatenated with the domains in the other protein as elements of the feature vector or use the domains to represent the feature vector with values based on if that domain was seen in 0, 1 or both the proteins of the PPI. Unlike previous studies the domain-domain interactions are used to represent elements of the feature vector in this study. By doing this the SVM is trained to recognize combinations of domain pairs that imply protein-protein interactions. As pointed out by Ben-Hur et al. [32] using features that characterize pairs of proteins such as domain-domain interactions are in concept the same as using a pairwise kernel such as that proposed by them. The feature vector corresponding to a protein pair (P, P') is therefore given by:

$$F(P, P') = [d_1\text{-}d_2, d_2\text{-}d_3, d_3\text{-}d_5, \ldots D\text{-}D_{in}] \qquad (3)$$

where d_i-d_j = domain pairs

and n = size of the subset of domain pairs (n=867 or n=5095 in this study)

These vectors were labeled as a +1 for positive PPI and -1 for negative PPI. The reciprocal protein pairs <p, p'> and <p', p> were removed from the training and test set creation since they are redundant.

5.2.3. Implementation

Several Python scripts were developed for pre- and post-processing of the data required for DomainGA. For example, pre-processing required scripts for creating and validating the positive and negative GSTD data; selecting DDI parameter sets based on occurrence counts; creating 867 and 5095 closed and inclusive sets; splitting the data in to 10-fold CV datasets for the 867-closed set, 867-inclusive set, 5095-closed set and 5095-inclusive sets; running the DomainGA optimization on all the 10 sets of the 867 & 5095 closed and inclusive sets; removing local maxima of GA optimizations and creating mean scores for DDI for each of the sets; creating feature vector representations using the mean, random, and other scores; and finally a script for automating the running the SVM implementation on the prepared data and collecting the statistics as shown in Figure 5.1.

SVMLight [75] was used as the SVM implementation in this study. Experiments are conducted with four kernel functions namely linear, sigmoidal, gaussian and polynomial. Similar to the observation by Ben-Hur et al., we also found that the linear kernel provides faster and better convergence than the other kernel methods despite the high dimensionality of the feature space; therefore all results presented in this paper were obtained using the linear kernel unless otherwise stated.

5.2.4. Assigning probability estimates to PPI

For a text example vector, the SVM outputs a score that provides the distance of that vector from a separating hyperplane. The class (interacting or non-interacting) of the test example can be deduced from the sign of the score. The magnitude of the SVM score by itself can only be used to compare predictions in the same set since it is relative to the training data used in the SVM. To be able to compare the SVM scores across training data, we need to take all the SVM scores across all training data and calibrated it into a conditional posterior probability estimate by using alternate measures such as by using the binning technique outlined by Drish [85]. In the binning technique the training examples are first rank ordered on the SVM scores and then divided into "n" equal sized bins. They recommend choosing the value of "n" experimentally such that the variance is minimal across bins. The fraction of the true positive training examples in each bin is used to represent the probability estimate of each test example that falls in the corresponding bin (gets an SVM score in that bin). For this study 10 bins were created and the fraction of true positive training examples calculated for each of them. As shown in Figure 5.3, the further the distance from the hyperplane the more likely the prediction being correct.

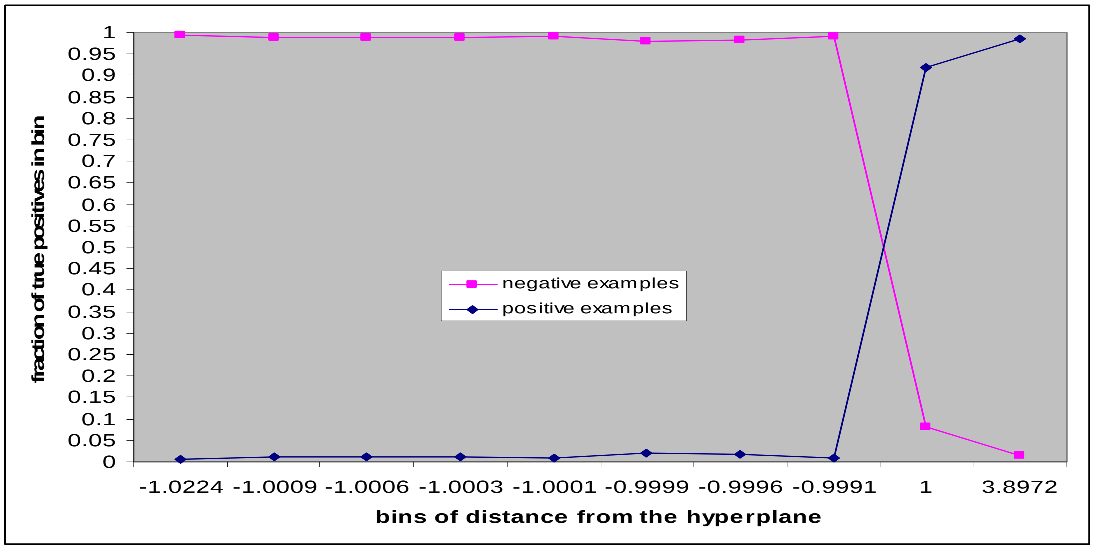


Figure 5.3: The distance from the hyperplane (SVM score) plotted against the probability estimates obtained from the binning technique. The greater the magnitude of the SVM score the higher the likelihood of it being in a particular class.

5.2.5. Performance Measures

Statistical validation using cross validation studies is a well recognized method for testing the usefulness of a prediction algorithm. 10-fold cross-validation studies were conducted to measure the performance of the different scoring schemes such as DomainGA scores, MLE method scores and Random scores used in the DomainSVM approach. In our evaluation the following measures of statistical validation were used.

$$\mathrm{Re}\,call(\mathrm{Re}) = TP\,/(TP{+}FN)$$
$$Specificity = TN\,/(TN{+}FP)$$
$$\mathrm{Pr}\,ecision(\mathrm{Pr}) = TP/(TP{+}FP)$$
$$F - measure = (2 * \mathrm{Pr} * \mathrm{Re})\,/(\mathrm{Pr} + \mathrm{Re})$$
$$ExplanationRatio = Accuracy = (TN{+}TP)\,/(TN + TP + FN + FP)$$

where TP, FN, TN and FN represent true positives, false negatives, true negatives and false negatives respectively. The F-measure is the harmonic mean of Precision and Recall and is an important measure to take into account the usefulness of a prediction algorithm if it is not dependent on the usage scenario.

5.3. Results and Discussion

5.3.1. Cross-Validation studies

10-fold cross validation studies were conducted to evaluate the performance of the DomainGA method, the results of which are summarized in Table 5.1 where the numbers are the average across the 10 sets. The table also shows the results obtained by using random scores (in the same range as DomainGA scores) for the DDI in the DomainSVM. It is important to note that this comparison is different from a true random comparison since it uses the information about the possible domain interactions amongst protein pairs to randomize the scores of those DDI instead of randomizing all the values in the feature vector.

Scores used in DomainSVM	**Accuracy** *closed set (inclusive set)*	**Precision** *closed set (inclusive set)*	**Recall(Sensitivity)** *closed set (inclusive set)*	**Specificity** *closed set (inclusive set)*	**F-measure** *closed set (inclusive set)*
DomainGA scores	98.9% (96.9%)	97.8% (95.0%)	96.8% (89.4%)	99.5% (98.8%)	97.3% (92.2%)
Random scores	96.0% (95.0%)	98.0% (96.0%)	80.0% (79.4%)	99.5% (99.2%)	88.0% (87.0%)
MLE scores (radial kernel)	96.9% (93.9%)	98.4% (96.2%)	85.9% (72.8%)	99.6% (99.3%)	91.7% (82.8%)

Table 5.1: Average of 10-fold cross-validation results using DomainSVM with different values (scores) for the elements in the feature vector. The DomainGA scores used are mean values across multiple runs of the algorithm. The data used was the closed and inclusive sets of 5095 parameters (DDI).

As can be seen from the table for the 5095 parameter closed set, the use of the DomainGA scores yielded a precision and recall of 97.8% and 96.8% respectively, with an average F-measure of 97.3% whereas the use of the random scores had much lower Sensitivity (Recall) of 80% with an F-measure of 88%.

The usefulness of domain-pair interaction scores obtained from DomainGA can also be established by comparison with the domain-pair interaction values obtained from the Maximum Likelihood Estimation method proposed by Deng et al. As shown in Table 5.1, the recall or sensitivity values of 10-fold cross-validation studies done on 5095 closed and inclusive sets are much lower than that obtained by the use of DomainGA scores. Although, the evaluation and validation of a machine learning based method is very difficult since it depends heavily on the choice of the data used for training and testing; 10-fold cross-validation (CV) is a popularly used technique for statistical evaluation of an approach and DomainSVM yielded very high prediction accuracies compared to those reported previously in a 10-fold CV study which demonstrates the validity of this approach.

5.3.2. Comparison of PPI predictions across approaches

PPI predictions obtained from the DomainSVM method were also compared with predictions obtained from the DomainGA method using statistics collected from the 10-fold cross-validation studies as shown in Figure 5.2. A careful analysis of the predictions incorrectly made by DomainGA and correctly predicted by DomainSVM showed that almost all the parameters involved had been assigned fuzzy values (mean values in the range of 4-5) by the DomainGA, which was one of the drawbacks of that approach. This observation leads us to believe that DomainSVM can more effectively detect combinations of domain-pairs interacting to predict PPI.

Algorithms	**Accuracy** *closed set* *(inclusive set)*	**Precision** *closed set* *(inclusive set)*	**Recall(Sensitivity)** *closed set* *(inclusive set)*	**Specificity** *closed set* *(inclusive set)*	**F-measure** *closed set* *(inclusive set)*
DomainSVM	98.9% (96.9%)	97.8% (95.0%)	96.8% (89.4%)	99.5% (98.8%)	97.3% (92.2%)
DomainGA	95.7% (95.7%)	96.0% (92.6%)	81.6% (85.7%)	99.0% (98.2%)	88.2% (89.0%)

Table 5.2: Average of 10-fold cross-validation results using different algorithms for predicting PPI. The data used was the closed and inclusive sets of 5095 DDI.

5.3.3. Testing the scalability

A limiting factor in an SVM based approach can be the dimensionality of the feature vector. Also, the right balance between the dimensionality and the amount of is training data is important to avoid over-fitting. To test the scalability of the DomainSVM approach, statistical comparison was conducted for the 867 and 5095 parameter set. As shown in Figure 5.3, there is very little difference in the accuracies of the 867 closed and inclusive sets versus the 5095 closed and inclusive sets showing that the DomainSVM performance does not deteriorate by increasing the dimensionality to up-to 5000 parameters and possibly higher.

Parameter set used in DomainSVM	**Accuracy** *closed set (inclusive set)*	**Precision** *closed set (inclusive set)*	**Recall(Sensitivity)** *closed set (inclusive set)*	**Specificity** *closed set (inclusive set)*	**F-measure** *closed set (inclusive set)*
867	98.0% (97.0%)	97.0% (95.0%)	95.0% (90.0%)	99.0% (98.8%)	96.0% (92.4%)
5095	98.9% (96.9%)	97.8% (95.0%)	96.8% (89.4%)	99.5% (98.8%)	97.3% (92.2%)

Table 5.3: Average of 10-fold cross-validation results using DomainSVM with different feature vector sizes. The data used was the closed and inclusive sets of 867 and 5095 DDI.

5.3.4. Functional analysis

Since the selection of the set of non-interacting PPI is a guess at the best based on co-localization information and the positive PPI set can also have inaccuracies, a functional comparison of proteins involved in PPI was conducted. The hypothesis is that interacting proteins share common functions or roles in a pathway. The Gene Ontology (GO) annotations database was used to obtain biological function information for the proteins of interest. The function information was compared at the leaf level (bottom-most in the hierarchy) and the semantic difference between "Is-A" and "Part-Of" was ignored for the purposes of this comparison. As summarized in Table 5.4, the percentage of true positives sharing GO annotations averages about 80% for all the 4 test cases being studied (867 & 5095 closed and inclusive sets).

	867-closed set	867-inclusive set	5095-closed set	5095-inclusive set

True positives	77.5%	81.3%	86.7%	85.9%
True negatives	2.1%	2.1%	2.1%	2.1%
False positives	33.3%	50.0%	54.5%	25.9%
False negatives	63.6%	53.4%	68.0%	58.0%

Table 5.4: Percentage functional similarity between proteins in a PPI in the predicted set

Comparison of biological functional annotations for new predictions (non-interacting in test set, but predicted to interact by the classifier) i.e. false positives of the test set showed that more than 50% of new predictions using DomainSVM involved proteins sharing common functions verses only 2% of the true negatives involved proteins with common functionality for the 5095 closed set. This difference between the functional similarity of the true negatives and the false positives for the other sets of 867-closed (33% vs. 2.1%) & inclusive sets (50% vs. 2%) and 5095-inclusive set (25% vs. 2%) were also significant.

CHAPTER SIX

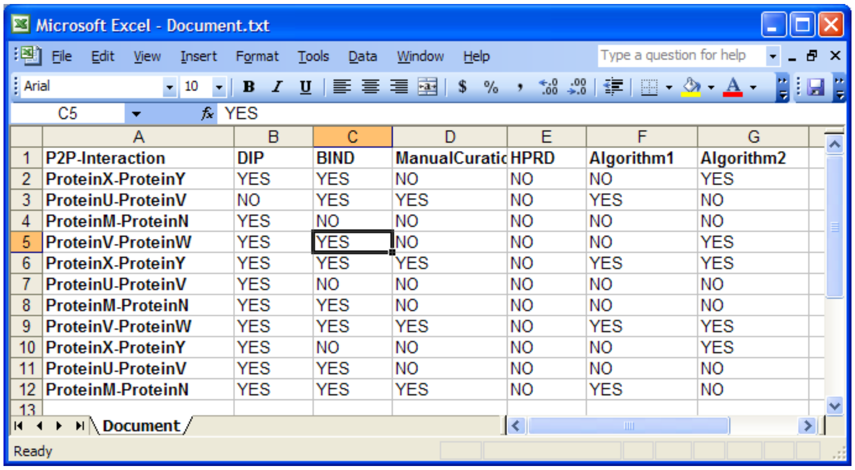
Visual Analytic Tool-CABIN

Abstract

The importance of understanding molecular interaction networks has fueled the development of numerous interaction data generation techniques, databases and prediction tools. However not all prediction tools and databases predict interactions with one hundred percent accuracy. Generation of high confidence interaction networks formulates the first step towards deciphering unknown protein functions, determining protein complexes and inventing drugs. The CABIN: Collective Analysis of Biological Interaction Networks software is an exploratory data analysis tool that enables comparison, analysis and integration of interactions evidence obtained from multiple sources, thereby increasing the confidence of computational predictions as well as validating experimental observations. CABIN has been written in Java^{TM} and is available as a plugin for Cytoscape – an open source network visualization tool. CABIN along with a user manual and tutorials is freely available for download at www.sysbio.org/dataresources/cabin.stm.

6.1. Introduction

Attaining a detailed understanding of the various biological networks in an organism lies at the core of the emerging discipline of systems biology. A precise description of the relationships formed between genes, mRNA molecules, and proteins is a necessary step toward a complete description of the dynamic behavior of an organism at the cellular level; and towards intelligent, efficient and directed modification of an organism. The importance of understanding such regulatory, signaling, and interaction networks has fueled the development of numerous in silico inference algorithms, as well as new experimental techniques and a growing collection of public databases that quantify PPIs based on their functional associations, phylogenetic profiles, sequence similarity, homology etc. These prediction tools as well as experimental techniques aim at assigning quantitative metrics to each interaction edge within such networks. In most cases, the evidence about these interactions can be incomplete and associated with uncertainty. As a result, human judgment and expertise has to be exercised while deriving a set of high-confidence interactions after assessing each source of data. The current methods for validating experimentally observed interactions involve checking interactions in public data sources like Prolinks [19], STRING [86], DIP [43], BIND [87], and Literature in a voting mechanism mostly done in a spreadsheet format such as Figure 6.1. There is a lack of computational tools that facilitate assignment of confidence to "sources of evidence" based on the reliability of the prediction method and dynamic cutoff assignment. Moreover the problem of combining evidence from multiple sources is compounded by typical identifier mapping problems such as redundancy and missing attributes etc.


Microsoft Excel - Document.txt
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
C5
YES
Document
Ready

	A	B	C	D	E	F	G
1	P2P-Interaction	DIP	BIND	ManualCuratic	HPRD	Algorithm1	Algorithm2
2	ProteinX-ProteinY	YES	YES	NO	NO	NO	YES
3	ProteinU-ProteinV	NO	YES	YES	NO	YES	NO
4	ProteinM-ProteinN	YES	NO	NO	NO	NO	NO
5	ProteinV-ProteinW	YES	YES	NO	NO	NO	YES
6	ProteinX-ProteinY	YES	YES	YES	NO	YES	YES
7	ProteinU-ProteinV	YES	NO	NO	NO	NO	NO
8	ProteinM-ProteinN	YES	YES	NO	NO	NO	NO
9	ProteinV-ProteinW	YES	YES	YES	NO	YES	YES
10	ProteinX-ProteinY	YES	NO	NO	NO	NO	YES
11	ProteinU-ProteinV	YES	YES	NO	NO	NO	NO
12	ProteinM-ProteinN	YES	YES	YES	NO	YES	NO
13							

Figure 6.1: Traditional spreadsheet approach of validating experimentally observed interactions.

CABIN is developed as a plugin to Cytoscape [88] – an open source network visualization tool to circumvent these problems by facilitating multiple evidence integration in a more intuitive manner. Multiple coordinated views within CABIN foster exploratory data analysis by users accommodating for expert domain knowledge. The functionalities available within CABIN maximize human perception and understanding of uncertain and complex data facilitating high quality human judgment with limited investment of the user's time. Predictive methods such as DomainGA and DomainSVM which assign confidence scores to interactions can be compared and further refined by combining them with confidence scores from other predictive methods in CABIN.

6.2. Research Design and Methodology

6.2.1. Visual Design

CABIN follows the basic information visualization principles of “focus+context” in which it displays the details of data at the focal point as well as the area around the focal point (the context) to help make sense of how the important information relates to the entire data structure. CABIN provides multiple coordinated viewers that represent the problem at multiple levels of abstraction as shown in Figure 6.2. A Scatter Plot Matrix viewer shows the relation amongst the different evidence-networks for the interactions of interest serving to facilitate the weight (confidence) assignment to each evidence-source. Matrix visualization within CABIN shows a heat map representation of the data values for all the evidence-sources in a tabular format. A Cytoscape network viewer shows the selected evidence-source in a node-edge format, providing the conventional node-edge kind of visualization for PPI networks. All the three viewers have functionality to make selections that are reflected in the other views. A filtering mechanism is provided to select subsets of interactions based on their evidence values shown in a histogram view. Finally a Weighted Evidence Viewer provides a view of the interactions with confidence based on the average weighted sum of its evidences. Along with these basic views, CABIN also has functionalities to search for interactions involving proteins of interest, mapping node attributes for all imported networks, functionality to drag and select interactions and functionality to save the selected or all interactions as a sub-network or in a tab delimited file on the local machine.

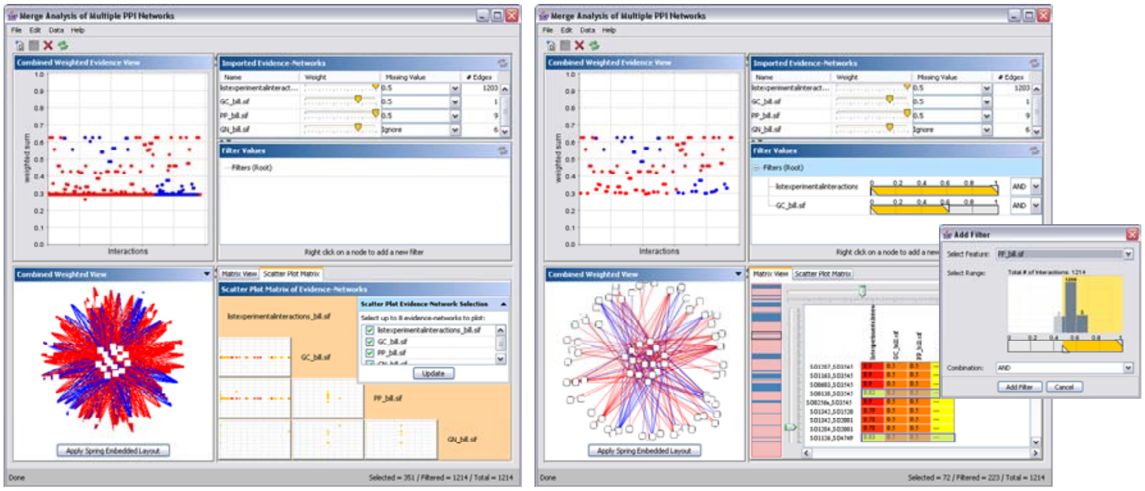

Merge Analysis of Multiple PPI Networks
File Edit Data Help
Combined Weighted Evidence View
Imported Evidence-Networks
Name
Weight
Missing Value
# Edges
listexperimentalinteract...
GC_bill.sif
PP_bill.sif
GN_bill.sif
Ignore
Filter Values
Filters (Root)
weighted sum
Interactions
Right click on a node to add a new filter
Combined Weighted View
Matrix View
Scatter Plot Matrix
Scatter Plot Matrix of Evidence-Networks
Scatter Plot Evidence-Network Selection
Select up to 8 evidence-networks to plot:
listexperimentalinteractions_bill.sif
GC_bill.sif
PP_bill.sif
Update
Apply Spring Embedded Layout
Done
Selected = 351 / Filtered = 1214 / Total = 1214
AND
Add Filter
Select Feature:
Select Range:
Total # of interactions: 1214
Combination:
Add Filter
Cancel
Selected = 72 / Filtered = 223 / Total = 1214

Figure 6.2 – Using CABIN to validate experimental interactions. The environment is loaded with experimentally observed interactions in Shewanella. The Prolinks database is used to create Phylogenetic Profile, Gene Cluster, Gene Neighborhood and Rosetta stone evidence networks for the same set of interactions. Exploratory analysis is being carried out by creating filters to select cutoffs for individual networks.

6.2.2. Implementation

CABIN has been written in Java^{TM} and is available for download as a Cytoscape plugin providing extended functionality to Cytoscape. CABIN makes use of publicly available libraries such as Colt [89], JFreeChart [90], jMatrixView [91] and BiSlider [92] to provide effective and rich visualizations. Once imported into CABIN, evidence networks are stored in a matrix model that keeps a list of the networks and their interactions. This model, provided by the high-performance Colt library, is backed by an optimized 2-dimensional sparse matrix which contains the confidence values of each interaction (row) of each network (column). These data values are visualized in multiple views as scatter plots (JFreeChart), as a heat map matrix representation (jMatrixView), and as Cytoscape networks. Each view references the matrix model and observes any changes in the model, allowing the views to update themselves when networks are imported, removed, or updated. Additionally, a view selection controller serves as an intermediary that notifies each registered view of any data selection events. Data values can be manipulated further using the histogram range slider interface facilitated by the BiSlider and JFreeChart libraries.

6.2.3. Features and Functionality

The functionalities provided within CABIN for visual analysis of multiple interaction networks include:

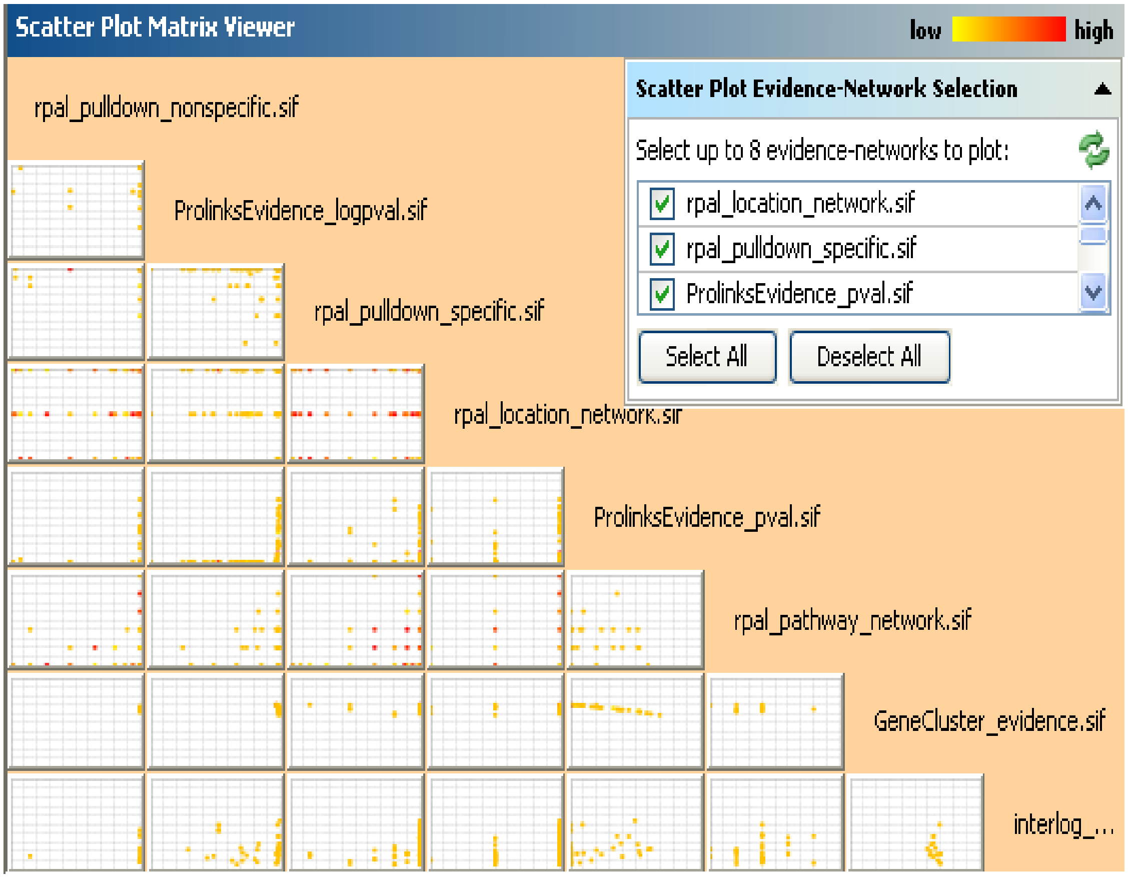
6.2.3.1 Network Import

Interaction networks inferred using the either the DomainGA, DomainSVM algorithms, or other publicly available inference methods based on phylogenetic profiling, gene neighborhood, gene cluster, homology etc can be imported into Cytoscape in the “sif” (simple interaction format) in which the first and the third columns represent the proteins in the interaction and the middle column represents the confidence value for that interaction based on the evidence source (inference method). Cytoscape also allows importing networks in the “gml” (graph markup language) and the “xggml” (XML format) formats. Networks imported into Cytoscape are CABIN compatible and can be imported into it by assigning a reliability score, or weight, based on the user-defined confidence in the evidence source. CABIN also has provisions to assign custom values to missing evidences for interactions. This missing value can be set to a value between 0 and 1, to the median value for that evidence network, or null.

6.2.3.2. Multiple Coordinated Viewers

6.2.3.2.1. Scatter Plot Matrix Viewer

The Scatter Plot Matrix viewer shows a matrix of scatter plots of the evidence-networks chosen in the *Scatter Plot Feature Selection Panel*. Each evidence-network is plotted against the other selected evidence-networks, showing a matrix with multiple scatter plots reflecting the relationship of all evidence-networks with all the others. A color gradient is used an indicator of density with yellow corresponding to lower density and red corresponding to higher density. An example of a scatter plot matrix between four selected evidence-networks is shown in Figure 6.3.


Scatter Plot Matrix Viewer
low
high
rpal_pulldown_nonspecific.sif
ProlinksEvidence_logpval.sif
rpal_pulldown_specific.sif
rpal_location_network.sif
ProlinksEvidence_pval.sif
rpal_pathway_network.sif
GeneCluster_evidence.sif
interlog_...
Scatter Plot Evidence-Network Selection
Select All
Deselect All

Select up to 8 evidence-networks to plot:

☑ rpal_location_network.sif

☑ rpal_pulldown_specific.sif

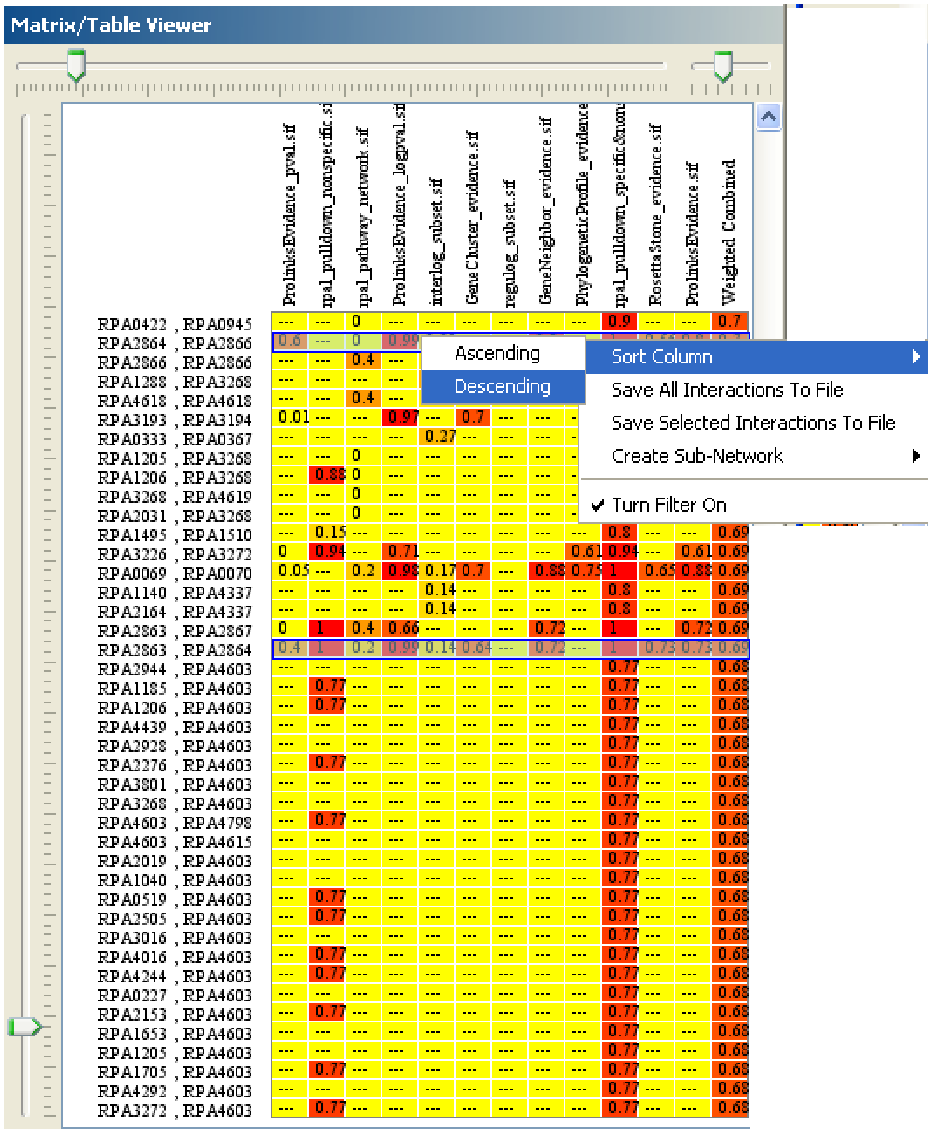
☑ ProlinksEvidence_pval.sif

Figure 6.3 Scatter Plot Matrix Viewer

A minimum of two and a maximum of eight evidence-networks can be selected for plotting at a time. The status bar shows an estimated completion time and the progress of the operation. Clicking on any scatter plot opens it up in a bigger window for a more detailed view of the plot. This viewer supports selection of points for corresponding information in the other views. The selection interactions can be saved as a new network within CABIN or exported to a local file.

6.2.3.2.2. Matrix Viewer

The Matrix Viewer is a heat map representation of the data values for all the evidence sources in a tabular format. Horizontal and vertical sliders are provided to expand the size of the columns and the rows for easy access to a large number of interactions. Sorting functionality facilitates easy selection of high confident interactions as shown in Figure 6.4. Continuous or discontinuous ranges of interactions can be selected by shift and control (ctrl) selecting rows in the matrix. Selection in this viewer is also coordinated with selection in the other viewers. Similarly, right-clicking anywhere in the view provides options to save selected or all interactions to a file or create a sub-network with the selected interactions for further exploration within CABIN.



	ProlinksEvidence_pval.sif	rpal_pulldown_nonspecific.si	rpal_pathway_network.sif	ProlinksEvidence_logpval.sif	interlog_subset.sif	GeneCluster_evidence.sif	regulog_subset.sif	GeneNeighbor_evidence.sif	PhylogeneticProfile_evidence	rpal_pulldown_specific&non	RosettaStone_evidence.sif	ProlinksEvidence.sif	Weighted Combined
RPA0422 , RPA0945	---	---	0	---	---	---	---	---	---	0.9	---	---	0.7
RPA2864 , RPA2866	0.6	---	0	0.99	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA2866 , RPA2866	---	---	0.4	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA1288 , RPA3268	---	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA4618 , RPA4618	---	---	0.4	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA3193 , RPA3194	0.01	---	---	0.97	---	0.7	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA0333 , RPA0367	---	---	---	---	0.27	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA1205 , RPA3268	---	---	0	---	---	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA1206 , RPA3268	---	0.88	0	---	---	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA3268 , RPA4619	---	---	0	---	---	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA2031 , RPA3268	---	---	0	---	---	---	---	---	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
RPA1495 , RPA1510	---	0.15	---	---	---	---	---	---	---	0.8	---	---	0.69
RPA3226 , RPA3272	0	0.94	---	0.71	---	---	---	---	0.61	0.94	---	0.61	0.69
RPA0069 , RPA0070	0.05	---	0.2	0.98	0.17	0.7	---	0.88	0.75	1	0.65	0.88	0.69
RPA1140 , RPA4337	---	---	---	---	0.14	---	---	---	---	0.8	---	---	0.69
RPA2164 , RPA4337	---	---	---	---	0.14	---	---	---	---	0.8	---	---	0.69
RPA2863 , RPA2867	0	1	0.4	0.66	---	---	---	0.72	---	1	---	0.72	0.69
RPA2863 , RPA2864	0.4	1	0.2	0.99	0.14	0.64	---	0.72	---	1	0.73	0.73	0.69
RPA2944 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1185 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1206 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4439 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA2928 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA2276 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA3801 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA3268 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4603 , RPA4798	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4603 , RPA4615	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA2019 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1040 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA0519 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA2505 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA3016 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4016 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4244 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA0227 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA2153 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1653 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1205 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA1705 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA4292 , RPA4603	---	---	---	---	---	---	---	---	---	0.77	---	---	0.68
RPA3272 , RPA4603	---	0.77	---	---	---	---	---	---	---	0.77	---	---	0.68

Figure 6.4 Matrix viewer

6.2.3.2.3. Graph/Cytoscape Viewer

The Cytoscape Viewer shows the selected evidence-network in a node-edge format. It provides options to view the different imported networks as well as the combined weighted network one-by-one in a node-edge representation as shown in Figure 6.5. This viewer has zoom-in and zoom-out functionality to delve-deeper into an area of interest. Interesting regions of the network can be selected, explored and exported after analysis.

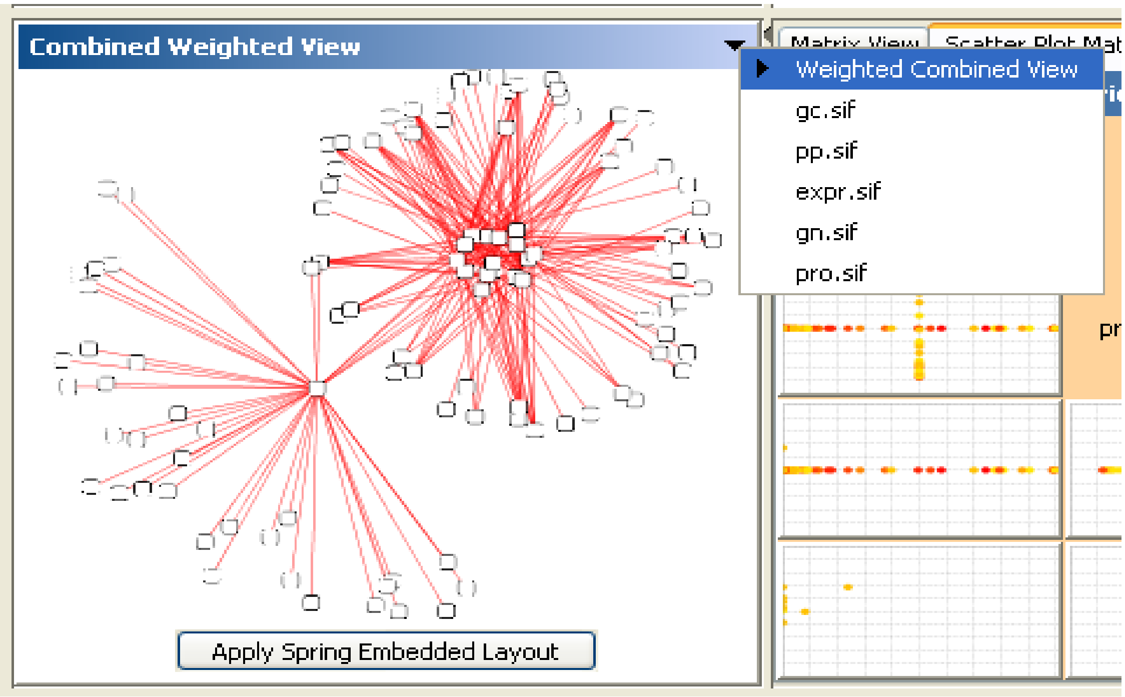


Figure 6.5. Combined Weighted View

6.2.3.2.4 Weighted Combined-Evidence Viewer

Each point in the Weighted Combined Evidence Viewer represents an interaction. This view shows a plot of the interactions with confidence values based on the weighted sum of all its evidences. The mouse pointer over a point shows the proteins involved in the interaction and its weighted sum. Multiple interactions can be selected by clicking and dragging over a rectangular area in the view as shown in Figure 6.6. The selected area is depicted by the red, outlined rectangular box. On release of the mouse button, the selected points are represented in blue color. Selecting points in this viewer will update the selections in the other viewers as well.

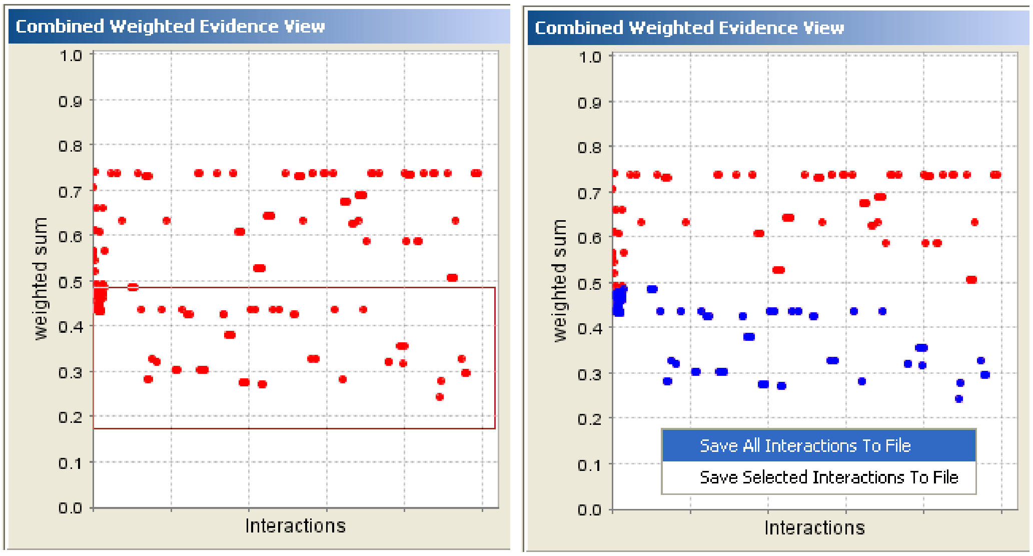
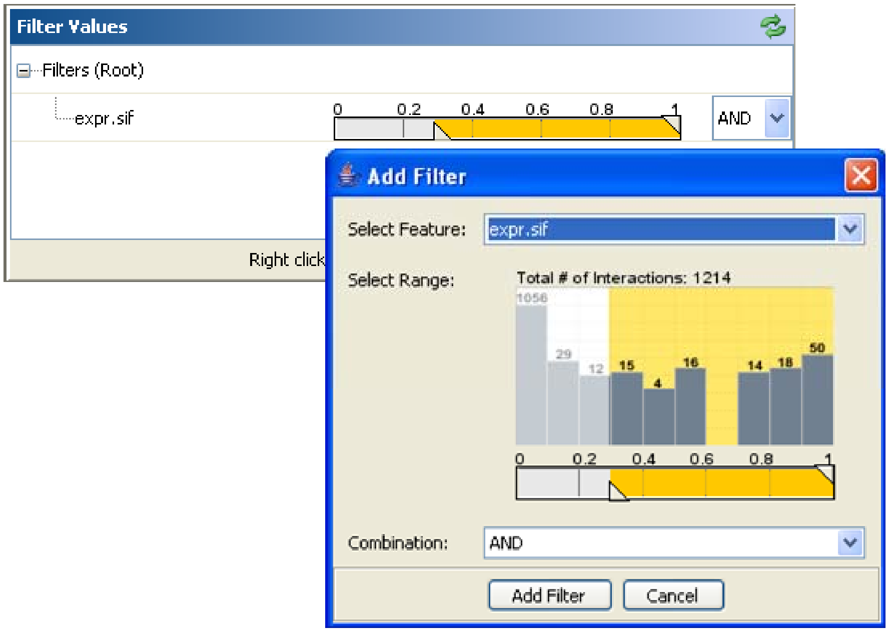

Combined Weighted Evidence View
weighted sum
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Interactions
Combined Weighted Evidence View
weighted sum
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Save All Interactions To File
Save Selected Interactions To File
Interactions

Figure 6.6 Combined Weighted Evidence Viewer

Selected or all interactions in this viewer can be saved to a file on the local machine by right-clicking anywhere within the view.

6.2.3.3. Filters

Once a network has more than a few hundred edges, it exploration becomes difficult especially in the conventional node-edge kind of representation. Moreover, to facilitate the selection of the "cutoff" parameter for confidence-value of an interaction from a public data source or an inference algorithm, CABIN has the functionality to assign dynamic filters to filter interactions in an evidence-network according to their confidence values. To add a filter for an evidence-network, CABIN shows a histogram distribution of the edge-values in that network along with a slider control to select the cutoff-value as shown in Figure 6.7. Edges displayed can also be restricted based on an OR relation or an AND relation amongst the networks; e.g., the combined network displayed (and possibly exported later) can be restricted to those edges that appear in the experimental network and that also appear in either the evidence network from DomainGA or the evidence network from DomainSVM (Experimental AND (DomainGA OR DomainSVM)). Once the filters are set and the update button is clicked, the views are updated based on interactions (edges) that pass the filters. The filtered set of edges can be saved as a new network within CABIN and assigned a confidence of its own.



Filter Values

Filters (Root)

expr.sif — AND

Add Filter

Select Feature: expr.sif

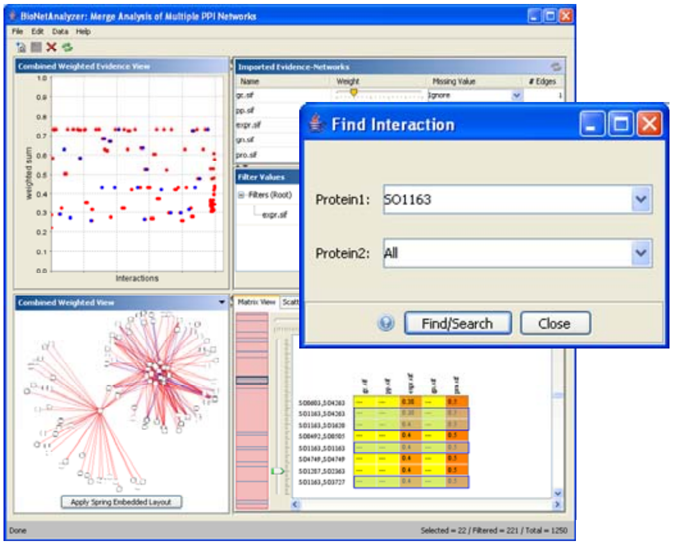
Select Range: Total # of Interactions: 1214

Combination: AND

Figure 6.7 Dynamic Filters

6.2.3.4 Find/Search

To facilitate selection of interactions involving a particular protein(s) of interest such as bait proteins, CABIN provides the search functionality as shown in Figure 6.8. Either the names of both the proteins can be entered (to select an interaction of interest) or one of the protein names can be entered (to select all interactions involving that protein). This regular expression based search tool facilitates searching for multiple proteins at one time as well such as using SO0*[0-100] to search for proteins from “SO0000” to “SO0100”.



Protein1: SO1163

Protein2: All

Done

Selected = 22 / Filtered = 221 / Total = 1250

Figure 6.8 Finding Interactions

6.2.3.5. Creating sub-networks

All the viewers support functionality to save the selected interactions as a new sub-network within CABIN. The source for the edge value (such as the combined-weighted score) and a weight need to be assigned to each such sub-network for comparison against the other imported networks. This functionality also allows selecting neighboring nodes of proteins of interest by specifying the depth of the network:

None - *only selected interactions*

Level 1 – *selected interactions and neighboring interactions one-hop away*

Level 2 – *selected interactions and neighboring interactions up to 2 hops away*

6.2.3.6. Exporting Results

CABIN provides the option to save all or selected interactions in a view to a file on your local machine. To save interactions to a file, right-click inside the view to bring up a popup menu with two options: save all interactions and save selection interactions as shown in

Figure 6.9. Choose one option from the menu depending on what you want to save. Choose the appropriate location from the file browser window that appears; specify a file name and click “Save” when finished.

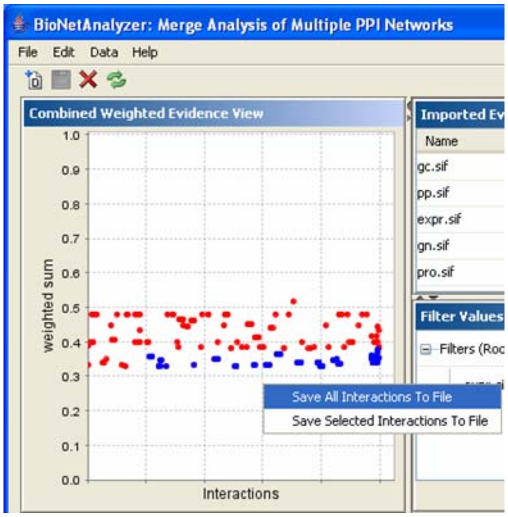

BioNetAnalyzer: Merge Analysis of Multiple PPI Networks
File
Edit
Data
Help
Combined Weighted Evidence View
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
weighted sum
Interactions
Imported Ev
Name
gc.sif
pp.sif
expr.sif
gn.sif
pro.sif
Filter Values
Filters (Roc
Save All Interactions To File
Save Selected Interactions To File

Figure 6.9 Save interactions to file

6.2.4. Data Sources

The following is a listing of the prediction algorithms and bioinformatics sources that can be used for creating evidence-networks for analysis and integration in CABIN:

- Prolinks: http://mysql5.mbi.ucla.edu/cgi-bin/functionator/pronav
- String: http://string.embl.de/
- Intact: http://www.ebi.ac.uk/intact/site/index.jsf
- DIP: http://dip.doe-mbi.ucla.edu/
- BIND: http://bond.unleashedinformatics.com/Action?
- HPRD: http://www.hprd.org/
- Agilent literature Search: Plug-in of Cytoscape

There are several inference algorithms that have been published in literature for inferring interactions amongst proteins from microarray data. These algorithms can be accessed from the Software Environment for Biological Network Inference (SEBINI) [93] developed at the Pacific Northwest National Laboratory (PNNL) and available at: https://www.emsl.pnl.gov/SEBINI/

- CLR: http://gardnerlab.bu.edu
- BANJO: http://www.cs.duke.edu/~amink/software/banjo/
- ARACNE [94]

This integration and comparative analysis of networks inferred from experimental data and computational predictive methods is one of the most unique and useful features of CABIN.

6.3. Results and Discussion

6.3.1. Case Study

In this section, a specific case study of using CABIN is discussed in a PPI network reconstruction project at the Pacific Northwest National Laboratory (PNNL). The MiPPI project [95] is a multi-year ORNL/PNNL collaboration to determine protein complexes and interaction networks in the bacterium *Rhodopseudomonas palustris* via mass spectrometry protein bait-prey experiments. CABIN forms the backbone of the exploratory analysis pipeline for this project. The downstream analysis of this project has three main steps: (1) use of the BEPro algorithm [96] to infer a PPI network from a set of 854 bait-prey experiments run at ORNL (2) obtaining evidence networks from bioinformatics sources (3) analysis of the resulting network in CABIN. Each of these steps is discussed in detail in the following sub-sections.

Step1: Using BEPro to infer PPI network from experimental data

Tandem affinity purification method is used to detect PPIs experimentally. 854 assays are conducted for R. palustris using this technique and then the BEPro : Bayesian Estimator of Protein-Protein Association Probabilities algorithm [97] is run on the assays. The BEPro program uses two algorithms designed to overcome errors pf pull-down experiments and produces a set of reliable PPIs, i.e., a network topology, by examining an annotated matrix across a set of assays. The following six parameters were set for the BEPro algorithm: (1) an estimate of the maximum number of prey proteins that can be observed with the given analysis method in R. palustris - set to 1,700, (2) the total number of proteins in the underlying proteome of R. palustris - set to 4,000, (3) the protein association score threshold, or cut point – set to 0, so that any positive value indicates "present", (4) the number of Monte Carlo simulations to perform – set to 50,000, (5) the maximum false positive rate, used to control the False Discovery Rate – set to 0.05, and (6) the posterior probability of protein-protein association threshold, i.e., the cutoff applied to the final association value to determine whether there is an interaction (a network edge to store) – set to 0.05. Using these parameter values, the BEPro specific LRT-Bayes algorithm returned 3,370 edges to store as the interaction network inferred from the set of 854 SEQUEST [98] runs 1,668 proteins from *R. palustris*.

Step2: Obtaining evidence networks from bioinformatics sources

For the set of proteins in the inferred network four evidence networks are obtained using the Phylogenetic Profile, Gene Cluster, Gene Neighborhood and Rosetta stone methods of the Prolinks database [25]. The details of these networks are provided in Table 6.1.

Name of Network	Source	Description
rpal_phylogenetic.sif	Phylogenetic profiles method from Prolinks database http://128.97.39.94/cgi_files/functionator/about.html	The Phylogenetic Profile network is created by using the phylogenetic profile method from the Prolinks database which uses the presence and absence of proteins across multiple genomes to detect functional linkages.
rpal_genecluster.sif	Gene Cluster method from Prolinks database http://128.97.39.94/cgi_files/functionator/about.html	The Gene Cluster method from the Prolinks database is used to create the gene cluster network which uses genome proximity to predict functional linkage.
rpal_rosettastone.sif	Rosetta Stone method from Prolinks database http://128.97.39.94/cgi_fil es/functionator/about.html	The Rosetta Stone network is created by using the Rosetta stone scores from the Prolinks database which uses a gene fusion event in a second organism to infer functional relatedness.
rpal_geneneighbor.sif	Gene Neighbor method from Prolinks database http://128.97.39.94/cgi_files/functionator/about.html	The Gene Neighbor method in the Prolinks database uses both gene proximity and phylogenetic distribution to infer linkage.

Table 6.1: Networks obtained from the Prolinks database

In addition, two evidence networks are obtained using protein information from the interolog and regulog methods [99] from the Bioverse database [100] with details provided in Table 5.2.

Name of Network	Source	Description
Rpal_interolog.sif	Interolog method of the Bioverse database http://bioverse.compbio.washington.edu/	The interolog method predicts an interaction between two proteins if they are both homologs of two proteins known to interact. Known protein interactions are gathered from databases of experimentally-determined PPIs (e.g. BIND, DIP) and PSI-BLAST is used to determine similarity between this set and all proteins in a target organism.
rpal_regulog.sif	Interolog method of the Bioverse database http://bioverse.compbio.washington.edu/	Regulogs are regulatory interactions inferred by homology. A regulog is predicted by determining similarity to a known transcription factor (TF) and the TF's target protein. Finally the nucleotide similarity in the upstream transcriptional promoter regions is determined and used to filter the regulog predictions: if there are similar promoter sequences then a regulog is predicted.

Table 6.2: Networks obtained from the Bioverse database

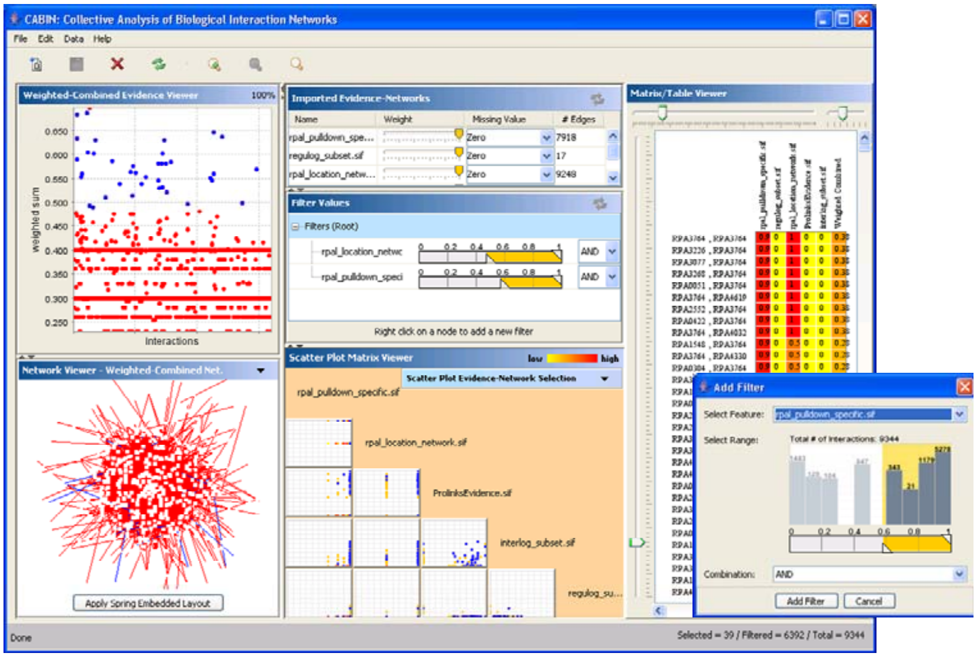
Another evidence network is obtained by utilizing the sub-cellular location information about the proteins with details described in Table 6.3.

Name of Network	Source	Description
rpal_location.sif	Sub-cellular localization information	The location network is created by looking at the sub-cellular compartment information of the two proteins in all possible interactions amongst the proteins in the pull-down experiments. The edge was assigned a value of 1 if both the proteins were localized in the same compartment and a value of 0 otherwise. If the compartment information for atleast one of the proteins was unavailable or the protein was localized in multiple compartments, then a value of 0.5 was given to the interactions involving those proteins.

Table 6.3: Networks created using sub-cellular localization information

Step3: Analysis of the resulting network in CABIN

Figure 6.10 shows the use of CABIN to validate experimental interactions inferred in this case study. These eight predicted networks (including the experimental network inferred using BEPro) are created as SIF files and imported into Cytoscape. They are then imported into CABIN by assigning a user-defined weight based on the confidence in the evidence source. For networks such as the cellular location network which has categorical information (tags), CABIN prompts the user to choose values for those tags at the time of import. Therefore values such as 1.0 for tag1, 0 for tag2 and 0.5 for tag3 can be chosen.



CABIN: Collective Analysis of Biological Interaction Networks

File Edit Data Help

Weighted-Combined Evidence Viewer 100%

Imported Evidence-Networks

Name	Weight	Missing Value	# Edges
rpal_pulldown_spe...		Zero	7918
regulog_subset.sif		Zero	17
rpal_location_netw...		Zero	9248

Filter Values

Filters (Root)

rpal_location_netwc AND

rpal_pulldown_speci AND

Right click on a node to add a new filter

Matrix/Table Viewer

	rpal_pulldown_specific.sif	regulog_subset.sif	rpal_location_network.sif	ProlinksEvidence.sif	interlog_subset.sif	Weighted Combined
RPA3764 , RPA3764	0.9	0	1	0	0	0.38
RPA3226 , RPA3764	0.9	0	1	0	0	0.38
RPA3077 , RPA3764	0.9	0	1	0	0	0.38
RPA3268 , RPA3764	0.9	0	1	0	0	0.38
RPA0051 , RPA3764	0.9	0	1	0	0	0.38
RPA3764 , RPA4619	0.9	0	1	0	0	0.38
RPA2552 , RPA3764	0.9	0	1	0	0	0.38
RPA0422 , RPA3764	0.9	0	1	0	0	0.38
RPA3764 , RPA4032	0.9	0	1	0	0	0.38
RPA1548 , RPA3764	0.9	0	0.5	0	0	0.28
RPA3764 , RPA4330	0.9	0	0.5	0	0	0.28
RPA0304 , RPA3764	0.9	0	0.5	0	0	0.28

Add Filter

Select Feature: rpal_pulldown_specific.sif

Select Range: Total # of interactions: 9344

Combination: AND

Selected = 39 / Filtered = 6392 / Total = 9344

0.650
0.600
0.550
0.500
0.450
0.400
0.350
0.300
0.250
weighted sum
Interactions
Network Viewer - Weighted-Combined Net.
Apply Spring Embedded Layout
Scatter Plot Matrix Viewer
low
high
Scatter Plot Evidence-Network Selection
rpal_pulldown_specific.sif
rpal_location_network.sif
ProlinksEvidence.sif
interlog_subset.sif
regulog_su...
0
0.2
0.4
0.6
0.8
1
1483
343
21
1179
5278
Add Filter
Cancel
Done

Figure 6.10 – CABIN use-case scenario. The CABIN software is used to validate experimental interactions for Rhodopseudomonas palustris obtained using tandem affinity purification technique bait-prey experiments. The interactions of interest are selected in blue.

All interaction values are normalized to a scale of 0-1 in CABIN at the time of import. As can be seen from Figure 6.10, there are 9,344 interactions imported into CABIN. The scatter plots show the correlation of the different interaction networks with respect to each other. It can be clearly seen that many inferred experimental interactions from the ORNL-PNNL MiPPI project (rpal_pulldown_specific.sif) have good agreement with the Prolinks predictions (prolinks_evidence. sif). Such agreement validates those experimental interactions, showing support from an independent data source. The regulog interaction network, on the other hand, has a low overlap with the other networks (points along the axis). This can be attributed to the obvious fact that PPIs (and/or interologs) are very different from regulatory interactions (and/or regulogs). Regulatory interactions act through an intermediate (the promoter region) so the transcription factor (TF) and TF target do not need to physically make contact. Only in (probably rare) cases where the protein produced from the TF gene target binds to its TF generally to inhibit its activity (an auto-regulatory loop) would you see both a protein-protein and regulatory interaction between the same pair of proteins. Two such interactions can be seen in this case; easily finding such interactions that may warrant further investigation is one advantage of CABIN.

Although the different views of the data give a deeper understanding of the multi-source data, the interpretation of an interaction network with more than a few hundred edges becomes difficult in a traditional network/graph like view. The use of filters in CABIN helps in sub-setting the data by changing the cutoff for the evidence networks dynamically. The interactions in Figure5.10 are filtered based on a value greater than 0.4 for the location network. Applying such a filter eliminated the interactions whose proteins are located in separate cellular compartments. Multiple filters can be added with AND/OR relationships between them, such as the filter applied on the inferred experimental network with a cutoff of 0.6. As shown in the status bar in CABIN, there are 6,392 interactions remaining after applying the filters in this case. The separation (points selected in blue) of the interactions can be clearly seen based on the combined confidence from all the evidence sources in the Weighted Scaling Viewer. These interactions are automatically selected in all the other viewers, showing their corresponding values in those views. Using the functionalities within CABIN, further data analysis can be carried out to validate the experimental interactions and, on conclusion of the analysis process, the high-confidence interactions can be saved in a local file.

6.3.2. Usage Scenarios

This section discusses some usage scenarios for CABIN.

6.3.2.1. Validating experimental observations

False predictions of an inferred network from experimental data can be validated by comparing it with predictions evidence obtained from other bioinformatics sources (such as Prolinks, DIP, BIND etc) or from other prediction algorithms (such as DomainGA, DomainSVM, Homology etc).

6.3.2.2. Network annotation or extension

A set of interactions involving detected proteins/genes can be extended or annotated by integrating evidence obtained from bioinformatics data sources.

6.3.2.3. Designing new experiments

Interaction evidence from several bioinformatics data sources can be integrated to construct a template/skeleton of a network and that can be used to determine the set of interesting proteins to be experimentally verified.

6.3.2.4. Comparing inference algorithms

Networks Inferred from experimental data using one of more inference algorithms such as CLR, ARACNE etc can be compared thereby evaluating the different inference algorithms and their parameter settings as well.

6.3.2.5. Exploration of dense networks

Interaction networks of more than a few hundred edges become difficult to explore and analyze in the conventional node-edge kind of representation. The coordinated viewers and filters in CABIN provide a better solution to explore these dense networks.

6.3.3. Availability

CABIN version 2.1 is available for download at the following location:

http://www.sysbio.org/dataresources/cabin.stm

The following are the terms of usage of the software:

6.3.3.1. Terms of Use

Notice: This computer software was prepared by Battelle Memorial Institute, hereinafter the Contractor, under Contract No. DE-AC05-76RL0 1830 with the Department of Energy (DOE). All rights in the computer software are reserved by DOE on behalf of the United States Government and the Contractor as provided in the Contract. NEITHER THE GOVERNMENT NOR THE CONTRACTOR MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LIABILITY FOR THE USE OF THIS SOFTWARE. This notice including this sentence must appear on any copies of this computer software.

6.3.3.2. Sample Data Release Policy

By using the sample PPI datasets provided within CABIN, the user agrees to the following: PNNL/ORNL is not responsible for errors contained in the PPI datasets, or for consequences arising from using the PPI datasets. Forthcoming publications from us will detail methods and controls used in the acquisition of data underlying the sample PPI datasets. The following acknowledgement must be included in any publications, presentations, reports, databases, websites, or data analyses that have used the CABIN PPI datasets: "PPI data were obtained from the Center for Molecular and Cellular Systems (mippi.ornl.gov) which is sponsored at Oak Ridge National Laboratory by the U.S. Department of Energy Office of

Biological and Environmental Research." The following paper needs to be cited in any research facilitated by CABIN: Singhal, M., Domico, K., "CABIN: Collective Analysis of Biological Interaction Networks", Computat. Biol. Chem. (2007), doi:10.1016/j.compbiolchem.2007.03.006. PNNL reserves the right to publish a description of the overall PPI datasets, and the methods used to obtain the contents of the dataset

CHAPTER SEVEN
CONCLUSIONS AND FUTURE WORK

High-throughput experimental methods to identify PPIs can be expensive and inaccurate, therefore computational methods can nicely complement experimental approaches. This thesis presents novel techniques for the prediction of PPI networks which provide a unique capability to map cellular pathways and their interconnectivities. The developed techniques use a combination of machine learning and visual analytic approaches. A genetic algorithm based approach called DomainGA assigns scores to DDIs and uses these scores to predict PPI. To accommodate for multiple domain-domain interactions, DomainSVM was presented, which uses the DDI scores obtained from DomainGA in an SVM based learning technique to increase the confidence in the predictions. Since several other predictive and experimental methods for PPI detection exist, each leveraging slightly different aspect of protein interactions, an exploratory visual analytic environment called CABIN is created to facilitate visualization, querying, comparison and hypothesis driven analysis of these interaction networks. The techniques developed in this thesis are expected to assist researchers in generating novel hypothesis and models.

DomainGA method predicts PPIs using the protein functional domain information and is tested for usefulness on the model organism *S. cerevisiae*. Because of the limitation imposed by the amount of available training data, in its current version only a small number of DDI pairs are selected as prediction parameters. As more experimental data become available, the reported scores can be improved and the domain parameter set can be expanded. Results with the larger 2466 and 5095 parameters show that this is possible when there is enough training data and that it is feasible to handle the added computational complexity. In addition to dealing with the PPI data specific to a specific organism, combining PPI data from multiple organisms can be used to create larger training and testing datasets. The encouraging results obtained in cross-verification tests where scores optimized using the yeast data were used to predict the human PPIs demonstrate that combining the data from multiple organisms will increase the predictive power of the DomainGA approach.

The possibility of false predictions is unavoidable in any computational method [13, 34, 38]. This may limit the usefulness of the computational PPI predictions to supplement the experimental observations. Keeping this in mind, the DomainGA is envisioned as a first step of a multi-tier approach to constructing PPIs. As a second step in a multi-tier approach the DomainSVM is presented which is an SVM based methodology for predicting PPI by utilizing domain-domain interaction information obtained from DomainGA. A careful analysis of PPI differently predicted in DomainGA and DomainSVM showed that those PPI involve fuzzy parameters (DDI with scores in the range of 4-6); fuzzy parameters being a limitation of the DomainGA method. Therefore it can be hypothesized that the detection of multiple domain interaction combinations as patterns lead to improved performance of DomainSVM over DomainGA. 10-fold statistical validation results showed that this approach yields better statistical results in particular the sensitivity or recall values than other existing methods. Moreover, comparison of biological functional annotations at the most detailed (leaf) level obtained from Gene Ontology (GO) database showed that 50% of new predictions using DomainSVM involved proteins sharing common functions verses only 2% of the true negatives involved proteins with common functionality. This also suggests the use of this approach to gain functional insight on proteins of unknown function involved in predicted interactions. As future work, including sub-cellular localization information in the feature vector can be expected to improve the prediction accuracy since physical interactions between proteins occur in specific compartments in the cell. However in that case care needs to be taken to validate the method with training data not created by looking at the sub-cellular localization information but by using alternate methods of creating the negative set such as those proposed by Ben-Hur et al. [47].

As it is based on fundamental structural information, the DomainGA and DomainSVM approaches can be used to create the potential PPIs, and the accuracy of the constructed interaction template can be improved later using complementary methods such as those based on literature search or location based evidence. Obtained explanation ratios during the reported test case studies clearly show that the false prediction rates of the obtained templates would be reasonably low and can be lowered even further with additional secondary tests conducted in the software tool CABIN. CABIN provides tools for visualizing and analyzing interactions data from multiple sources of evidence. This tool helps the user investigate their data in much greater detail than what is possible in the conventional spreadsheets. CABIN also provides the ability to integrate the domain expertise into the analysis process by being able to assign confidence values to sources of evidence and dynamically changing cutoffs for filtering interactions. Not only is such a tool useful for validating experimental observations, but is also useful in designing experimental studies based on computational prediction of highly confident interactions. Future work involves refining the weights assignment process by providing default weights based on statistical reliability of the features; normalizing discrete or rank based data effectively; and providing advanced algorithms for creating the weighted combined view. CABIN has more than 150 users till date and is being used in the analysis pipeline of large-scale Genomes to Life project at PNNL to find PPI networks in bacteria, as discussed in section 6.3.1. In addition, CABIN is being employed to find interactions in mass spectrometry data for studying the insulin signaling pathway in mouse in the Environmental and Molecular Sciences Laboratory (EMSL) at PNNL; in exploring experimentally observed interactions in the human proteome by scientists at the Harvard Medical School; as well as in the analyses of inferred networks from large sized micro-array data sets at North Carolina State University at Charlotte. Finally choosing the right data sources and their weight assignment is crucial to conduct an un-biased analysis. It is important to realize that CABIN does not give a final answer; it just helps see interesting aspects in the data which need to be experimentally verified.

In terms of limitations of these methods, it is important to note that while extracting the protein domains, it has been implicitly assumed that the variations in the amino acid composition of the same domain type among proteins do not alter the domain's interaction patterns. As amino acid substitutions may impact complex formation affinities, disregarding the exact sequence of the functional domains may lead to failures in some cases. Inclusion of such local structural characteristics can be very useful in predicting the effects of mutations [101] and alternate splicing events [102]. Even though the necessary computational extension to include the local amino acid sequence dependence is straightforward, inclusion of the amino acid composition of the functional domains into the interaction score scheme would require a combinatorial increase in the needed training dataset sizes. Such generalizations are currently impractical, but they will be included in future studies as such details are warranted.

BIBLIOGRAPHY

1. Alberts, B., et al., *Macromolecules: structure, shape, and function.* Molecular Biology of the Cell. 2nd edn., Garland, New York, 1989.

2. Tucker, C.L., J.F. Gera, and P. Uetz, *Towards an understanding of complex protein networks.* Trends Cell Biol, 2001. **11**(3): p. 102-6.

3. Dove, A., *Proteomics: translating genes into products?* Nature Biotechnology, 1999. **17**: p. 233-236.

4. Uetz, P., et al., *A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae.* Nature, 2000. **403**(6770): p. 623-7.

5. Ito, T., et al., *A comprehensive two-hybrid analysis to explore the yeast protein interactome.* Proc Natl Acad Sci U S A, 2001. **98**(8): p. 4569-74.

6. Ho, Y., et al., *Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry.* Nature, 2002. **415**(6868): p. 180-3.

7. Gavin, A.C., et al., *Functional organization of the yeast proteome by systematic analysis of protein complexes.* Nature, 2002. **415**(6868): p. 141-7.

8. Zhu, H., et al., *Global analysis of protein activities using proteome chips.* Science, 2001. **293**(5537): p. 2101-5.

9. Tong, A.H., et al., *A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules.* Science, 2002. **295**(5553): p. 321-4.

10. Tong, A.H.Y., et al., *Systematic Genetic Analysis with Ordered Arrays of Yeast Deletion Mutants.* Science, 2001. **294**(5550): p. 2364-2368.

11. Hollingsworth, R. and J.H. White, *Target discovery using the yeast two-hybrid system* Drug Discovery Today: TARGETS 2004. **3**(3): p. 97-103.

12. von Mering, C., et al., *Comparative assessment of large-scale data sets of protein-protein interactions.* Nature, 2002. **417**(6887): p. 399-403.

13. Legrain, P., J. Wojcik, and J.M. Gauthier, *Protein--protein interaction maps: a lead towards cellular functions.* Trends Genet, 2001. **17**(6): p. 346-52.

14. Sprinzak, E., S. Sattath, and H. Margalit, *How reliable are experimental protein-protein interaction data?* J Mol Biol, 2003. **327**(5): p. 919-23.

15. Enright, A.J., et al., *Protein interaction maps for complete genomes based on gene fusion events.* Nature, 1999. **402**(6757): p. 86-90.

16. Dandekar, T., et al., *Conservation of gene order: a fingerprint of proteins that physically interact.* Trends Biochem Sci., 1998. **23**(9): p. 324-328.

17. Marcotte, E.M., et al., *Detecting protein function and protein-protein interactions from genome sequences.* Science, 1999. **285**(5428): p. 751-753.

18. Bock, J.R. and D.A. Gough, *Predicting protein--protein interactions from primary structure.* Bioinformatics, 2001. **17**(5): p. 455-60.

19. Bowers, P., et al., *Prolinks : a database of protein functional linkages derived from coevolution* Genome Biology 2004. **5**(5): p. R35.

20. Pellegrini, M., et al., *Assigning protein functions by comparative genome analysis: Protein phylogenetic profiles* Biochemistry, 1999. **96**(8): p. 4285-4288.

21. Chen, L., et al., *Inferring protein interactions from experimental data by association probabilistic method.* Proteins, 2006. **62**(4): p. 833-7.

22. Deng, M., et al., *Inferring domain-domain interactions from protein-protein interactions.* Genome Res, 2002. **12**(10): p. 1540-8.

23. Gomez, S.M., S.H. Lo, and A. Rzhetsky, *Probabilistic prediction of unknown metabolic and signal-transduction networks.* Genetics, 2001. **159**(3): p. 1291-8.

24. Gomez, S.M., W.S. Noble, and A. Rzhetsky, *Learning to predict protein-protein interactions from protein sequences.* Bioinformatics, 2003. **19**(15): p. 1875-81.

25. Guimarães, K.S., et al., *Predicting domain-domain interactions using a parsimony approach.* Genome Biology, 2006. **7**(11): p. R104.

26. Han, D.S., et al., *PreSPI: design and implementation of protein-protein interaction prediction service system.* Genome Inform Ser Workshop Genome Inform, 2004. **15**(2): p. 171-80.

27. Kim, W.K., J. Park, and J.K. Suh, *Large scale statistical prediction of protein-protein interaction by potentially interacting domain (PID) pair.* Genome Inform Ser Workshop Genome Inform, 2002. **13**: p. 42-50.

28. Riley, R., et al., *Inferring protein domain interactions from databases of interacting proteins.* Genome Biol, 2005. **6**(10): p. R89.

29. Sprinzak, E. and H. Margalit, *Correlated sequence-signatures as markers of protein-protein interaction.* J Mol Biol, 2001. **311**(4): p. 681-92.

30. Wojcik, J., I.G. Boneca, and P. Legrain, *Prediction, assessment and validation of protein interaction maps in bacteria.* J Mol Biol, 2002. **323**(4): p. 763-70.

31. Wojcik, J. and V. Schachter, *Protein-protein interaction map inference using interacting domain profile pairs.* Bioinformatics, 2001. **17 Suppl 1**: p. S296-305.

32. Ben-Hur, A. and W.S. Noble, *Kernel methods for predicting protein-protein interactions.* Bioinformatics, 2005. **21 Suppl 1**: p. i38-i46.

33. Dohkan, S., A. Koike, and T. Takagi, *Prediction of Protein-Protein Interactions Using Support Vector Machines.* Fourth IEEE Symposium on Bioinformatics and Bioengineering, 2004: p. 576.

34. Jansen, R., et al., *A Bayesian networks approach for predicting protein-protein interactions from genomic data.* Science, 2003. **302**(5644): p. 449-53.

35. Lu, L., et al., *Assessing the limits of genomic data integration for predicting protein networks.* Genome Research, 2005. **15**: p. 945-953.

36. Ng, S.K., Z. Zhang, and S.H. Tan, *Integrative approach for computationally inferring protein domain interactions.* Bioinformatics, 2003. **19**(8): p. 923-9.

37. Valencia, A. and F. Pazos, *Computational methods for the prediction of protein interactions.* Curr Opin Struct Biol, 2002. **12**(3): p. 368-73.

38. Martin, S., D. Roe, and J.-L. Faulon, *Predicting protein-protein interactions using signature products.* Bioinformatics, 2005. **21**(2): p. 218-226.

39. Gomez, S.M. and A. Rzhetsky, *Towards the prediction of complete protein--protein interaction networks.* Pac Symp Biocomput, 2002: p. 413-24.

40. Rhodes, D.R., et al., *Probabilistic model of the human protein-protein interaction network.* Nat Biotechnol, 2005. **23**(8): p. 951-9.

41. Xenarios, I., et al., *DIP: The Database of Interacting Proteins: 2001 update.* Nucleic Acids Res, 2001. **29**(1): p. 239-41.

42. Xenarios, I., et al., *DIP: the database of interacting proteins.* Nucleic Acids Res, 2000. **28**(1): p. 289-91.

43. Xenarios, I., et al., *DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions.* Nucleic Acids Res, 2002. **30**(1): p. 303-5.

44. Bateman, A., et al., *The Pfam protein families database.* Nucleic Acids Res, 2002. **30**(1): p. 276-80.

45. Bateman, A., et al., *The Pfam protein families database.* Nucleic Acids Res, 2000. **28**(1): p. 263-6.

46. Bateman, A., et al., *The Pfam protein families database.* Nucleic Acids Res, 2004. **32**(Database issue): p. D138-41.

47. Ben-Hur, A. and W.S. Noble, *Choosing negative examples for the prediction of protein-protein interactions.* BMC Bioinformatics, 2006. **7 Suppl 1**: p. S2.

48. Bader, J.S., *Greedily building protein networks with confidence.* Bioinformatics, 2003. **19**(15): p. 1869-74.

49. Hoffmann, R. and A. Valencia, *Protein interaction: same network, different hubs.* Trends Genet, 2003. **19**(12): p. 681-3.

50. Jansen, R., et al., *Integration of genomic datasets to predict protein complexes in yeast.* J Struct Funct Genomics, 2002. **2**(2): p. 71-81.

51. Lee, I., et al., *A probabilistic functional network of yeast genes.* Science, 2004. **306**(5701): p. 1555-8.

52. Guldener, U., et al., *MPact: the MIPS protein interaction resource on yeast.* Nucleic Acids Res, 2006. **34**(Database issue): p. D436-41.

53. Leslie, C. and E. Eskin, *Mismatch string kernels for discriminative protein classification.* Bioinformatics, 2003. **1**(1): p. 1-10.

54. *The Gene Ontology Consortium. Gene Ontology: tool for the unification of biology.* Nature Genetics, 2000. **25**: p. 25-29.

55. Ng, S.K., et al., *InterDom: a database of putative interacting protein domains for validating predicted protein interactions and complexes.* Nucleic Acids Res, 2003. **31**(1): p. 251-4.

56. *MIPS Resource [http://mips.gsf.de/].*

57. Mewes, H.W., et al., *MIPS: analysis and annotation of proteins from whole genomes.* Nucleic Acids Res, 2004. **32**(Database issue): p. D41-4.

58. Mulder, N.J., et al., *InterPro: an integrated documentation resource for protein families, domains and functional sites.* Brief Bioinform, 2002. **3**(3): p. 225-35.

59. Mulder, N.J., et al., *InterPro, progress and status in 2005.* Nucleic Acids Res, 2005. **33**(Database issue): p. D201-5.

60. Sigrist, C.J., et al., *PROSITE: a documented database using patterns and profiles as motif descriptors.* Brief Bioinform, 2002. **3**(3): p. 265-74.

61. Mi, H., et al., *The PANTHER database of protein families, subfamilies, functions and pathways.* Nucleic Acids Res, 2005. **33**(Database issue): p. D284-8.

62. Attwood, T.K., et al., *PRINTS and its automatic supplement, prePRINTS.* Nucleic Acids Res, 2003. **31**(1): p. 400-2.

63. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic Acids Res, 1997. **25**(17): p. 3389-402.

64. *Uetz dataset [http://dip.doe-mbi.ucla.edu/dip/Download.cgi].*

65. Deane, C.M., et al., *Protein Interactions: Two Methods for Assessment of the Reliability of High Throughput Observations.* Mol Cell Proteomics, 2002. **1**(5): p. 349-356.

66. *The Cancer Cell Map [http://cancer.cellmap.org/cellmap/home.do].*

67. Peri, S., *Development of human protein reference database as an initial platform for approaching systems biology in humans.* . Genome Research, 2003. **13**: p. 2363-2371.

68. Keskin, O., et al., *A new, structurally nonredundant, diverse data set of protein-protein interfaces and its implications.* Protein Sci, 2004. **13**(4): p. 1043-55.

69. Lin, N., et al., *Information assessment on predicting protein-protein interactions.* BMC Bioinformatics, 2004. **5**: p. 154.

70. Zhang, L.V., et al., *Predicting co-complexed protein pairs using genomic and proteomic data integration.* BMC Bioinformatics, 2004. **5**: p. 38.

71. Mi H, L.-U.B., Loo R, Kejariwal A, Vandergriff J, Rabkin S, Guo N, Muruganujan A, Doremieux O, Campbell MJ, Kitano H, Thomas PD., *The PANTHER database of protein families, subfamilies, functions and pathways.* Nucleic Acids Res., 2005. **33(Database issue)**: p. D284-8.

72. Singhal, M. and H. Resat, *A domain-based approach to predict protein-protein interactions.* BMC Bioinformatics, 2007. **8**(199).

73. Mott, R., et al., *Predicting protein cellular localization using a domain projection method.* Genome Res, 2002. **12**(8): p. 1168-74.

74. Guimaraes, K.S., et al., *Predicting domain-domain interactions using a parsimony approach.* Genome Biol, 2006. **7**(11): p. R104.

75. *SVMLight [http://svmlight.joachims.org/].*

76. Vapnik, V.N., *Statistical Learning Theory. Adaptive and learning systems for signal processing, communications, and control*, ed. Wiley. 1998, New york: Wiley.

77. Vapnik, V.N., *The nature of Statistical Learning Theory*, ed. Springer. 1995, New York: Springer.

78. Ogul, H. and E.U. Mumcuoglu, *A discriminative method for remote homology detection based on n-peptide compositions with reduced amino acid alphabets.* Journal of Molecular Biology, 2006. **284**(4): p. 1202-1210.

79. Shah, A.R., et al., *Integrating Subcellular Location for Improving Machine Learning Models of Remote Homology Detection in Eukaryotic Organisms.* Computational Biology and Chemistry 2007. **31**(2): p. 138-142.

80. Smith, T. and M. Waterman, *Identification of common molecular subsequences.* Journal of Molecular Biology, 1981. **147**: p. 195-197.

81. Webb-Robertson, B.-J.M., C. Oehmen, and M. Matzke, *SVM-BALSA: Remote homology detection based on Bayesian sequence alignment.* Computational Biology and Chemistry, 2005. **29**: p. 440-443.

82. Lu, Z., et al., *Predicting Subcellular Localization of Proteins using Machine-Learned Classifiers.* Bioinformatics, 2004. **20**(4): p. 547-556.

83. Sarda, D., et al., *pSLIP: SVM based protein subcellular localization prediction using multiple physicochemical properties.* BMC Bioinformatics, 2005. **6**(152).

84. Yu, C.-S., C.-J. Lin, and J.-K. Hwang, *Predicting subcellular localization of protein for Gram-negative bacteria by support vector machines based on n-peptide compositions.* Protein Science, 2004. **13**: p. 1402-1406.

85. Drish, J. (2001) *Obtaining calibrated probability estimates from support vector machines*. **Volume**,

86. von Mering, C., et al., *STRING 7--recent developments in the integration and prediction of protein interactions*. Nucleic Acids Research, 2007. **35**(Database issue): p. D358-D362.

87. Bader, G.D., D. Betel, and C.W. Hogue, *BIND: the Biomolecular Interaction Network Database*. Nucleic Acids Res, 2003. **31**(1): p. 248-50.

88. Shannon, P., et al., *Cytoscape: a software environment for integrated models of biomolecular interaction networks*. Genome Research, 2003. **13**(11): p. 2498-2504.

89. *Colt Library [http://dsd.lbl.gov/~hoschek/colt]*.

90. *JFreeChart Library [http://www.jfree.org/jfreechart]*.

91. *JMatrixView Library [jmatrixview.sourceforge.net]*.

92. *BiSlider Library [https://bislider.dev.java.net]*.

93. Taylor, R.C., et al., *SEBINI: Software Environment for BIological Network Inference*. Bioinformatics, 2006. **21**: p. 2706-2708.

94. Margolin, A.A., *Reverse engineering cellular networks*. Nature Protocols, 2006. **1**(2): p. 663-672.

95. *MiPPI project [http://mippi.ornl.gov/]*.

96. Sharp, J.L., et al., *Statistically Inferring Protein-Protein Associations with Affinity Isolation LC-MS/MS Assays*. Journal of Proteome Research, 2007. **6**(9): p. 3788-3795.

97. *BEPro [http://www.pnl.gov/statistics/bepro3/index.htm]*.

98. *SEQUEST [http://fields.scripps.edu/sequest/]*.

99. Yu, H., et al., *Annotation transfer between genomes: protein-protein interologs and protein-DNA regulogs.* Genome Res., 2004. **14**(6): p. 1107-18.

100. *Bioverse [http://bioverse.compbio.washington.edu/].*

101. Wang, Z., Moult, J., *SNPs, protein structure, and disease.* Hum Mutat, 2001. **17(4)**: p. 263-70.

102. Resch, A., et al., *Assessing the impact of alternative splicing on domain interactions in the human proteome.* J Proteome Res, 2004. **3**(1): p. 76-83.