A method based on multiple SVMs for a comprehensive prediction of protein-protein interactions

複数のSVMsに基づく網羅的タンパク質間相互作用予測手法

Shinsuke Dohkan

道菅　紳介

A Dissertation Presented
by
Shinsuke Dohkan

Submitted to
the Graduate School of Frontier Sciences of the University of Tokyo
in partial ful llment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2006

Department of Computational Biology

Abstract

The knowledge on physical protein-protein interactions provides us with many clues for the understanding of living organisms as integrated systems. However, the protein-protein interactions identified so far cover only a small fraction of the total number of interactions occurring *in vivo*, and it is suggested that the low coverage sometimes lead to misunderstanding of the biological systems. To facilitate researchers to detect true protein-protein interactions and frame hypotheses efficiently, this thesis proposes a computational method for predicting protein-protein interactions in yeasts, mice, and humans.

Many previous methods used the information on protein domains and had better performances than traditional methods based on comparative analyses of genomic sequences. These methods have, however, two major defects that limit their performances and applicabilities: the assumption of domain independence and the difficulty in integrating other protein features. We propose a method based on Support Vector Machines (SVMs) that can address both of the two problems.

We first examined the performance of our method in predicting yeast protein-protein interactions. As a result, the highest F-measure of 0.788 was obtained by combining the features "domains," "amino acid compositions," and "subcellular localizations," which was more accurate than the predictions reported previously. We then found that our method could predict 58.6% of likely interactions in a dataset produced by yeast two-hybrid systems, and that newly predicted interactions tended to share similar functions between two proteins. The next challenging problem on which few previous works have focused is to predict interactions between mammalian proteins. Our SVMs trained on human protein pairs achieved an F-measure of 0.776 in predicting interactions in humans and an F-measure of 0.765 in predicting interactions in mice, indicating that our method can be applied to predicting interactions between mammalian proteins.

The performance must be further improved for constructing a hypothetical protein-protein interaction map computationally, because even a good classifier yields a huge number of false positives if the input data is all pairs of proteins in a given organism. Thinking that the negatives used in previous

studies cannot adequately represent all the negatives that need to be taken into account, we developed a method based on multiple SVMs for constructing hypothetical interaction maps of yeasts and humans. We found that the performance improved as we increased the number of SVMs and that, if more than one CPU is available, an approach using multiple SVMs was useful not only for improving the performance of classifiers but also for reducing the time required for training them. This multiple-SVM-based approach can also be applied to assessing the reliability of a dataset generated by high-throughput systems such as yeast two-hybrid.

One can predict interactions between proteins of interest by using our Web-based service. The server, which we call PIPS, provides a way of applying our multiple-SVM-based method to predicting physical protein-protein interactions in yeast, mice, and humans. The predicted protein-protein interactions and resulting maps will serve as an important resource for inferring and identifying protein functions, functional modules, or even the mechanism of gene evolution.

Acknowledgements

This thesis could never have been written without the support and help of several people. First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Toshihisa Takagi, who provided me with a great environment in which I could develop logical thinking, creativity, coping skills, leadership skills, and many other skills that a researcher should have. I owe a great debt to my thesis advisor Dr. Asako Koike, who is a visiting Associate Professor at Graduate School of Information Science and Technology at University of Tokyo, for her critical remarks, guidance, and suggestions on my research.

I am deeply grateful to the members of the thesis committee, Professor Kiyoshi Asai, Professor Takashi Ito, Associate Professor Akio Kitao, and Associate Professor Akihoro Nakaya. Their comments and suggestions were extremely helpful to improve the quality of the thesis.

Deep thanks go to Dr. Yasunori Yamamoto and Dr. Hideki Noguchi, who inspired me through seminar discussions and personal communications. The following staff of the Takagi lab warrant special mention: Ms. Naoko Tomioka, Ms. Yuko Muto, and Mr. Nobuyuki Okada.

This work was supported in part by Grant-in-Aid for Scientific Research on Priority Areas "Genome Information Science" and "Systems Genomics", both from the Japanese Ministry of Education, Culture, Sports, Science, and Technology.

Finally, I am deeply indebted to Hitachi Ltd. for providing me with an opportunity to study computational biology and for funding my education.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

One of the major issues in the postgenomic era is to uncover the whole physical protein-protein interactions in an organism. The information on protein-protein interactions provides us with many clues for the systematic understanding of biological systems. For example, the accumulation of interaction data has enabled us to infer the functions of uncharacterized proteins systematically from the known functions of their interaction partners [21, 29, 43, 63, 73]. Other important biological knowledge can also be extracted from the protein-protein interaction data. The existence of the scale-free property in a protein-protein interaction network suggests that whereas most proteins interact with few partners, some proteins interact with many partners [24, 36, 44, 45]. These highly connected proteins, called hubs, play a central role in mediating interactions among less connected proteins and are three times more likely to be essential [36]. Other researches showed that the motifs or modules in an interaction network could be a useful unit for the understanding of the complicated interaction network. The fully connected motifs (cliques consisted of less than or equal to five nodes) tend to be evolutionary conserved and are likely to be protein complexes [79]. The modules obtained after the complete removal of the hub proteins that are differently expressed from their interaction partners well correspond to the functional units in biological system [27]. Protein-protein interaction data have also been used for inferring the mechanism of gene evolution [51] and for revealing the variety and constancy of interaction networks among organisms [64].

It is important to remember, however, that the protein-protein interactions identified so far cover only a small fraction of the total number of interactions

occurring *in vivo* [3, 74, 75], and are considered to be contaminated by false positives [74]. Low coverage of interactions leads to misunderstanding of the biological systems. For example, recent research suggested that the true topology of protein-protein interaction networks cannot be determined unless the coverage is increased through further experimentation [28]. Obviously, false positives in a dataset also lead to erroneous conclusions.

Biological experiments for the identification of comprehensive protein-protein interactions are costly, labor-intensive, and time-consuming. Genome sequencing projects have revealed the complete DNA sequences of 426 organisms and the draft assembly of 357 organisms, and 502 projects are still in progress [78]. Unfortunately, it seems impossible to set up experiments for elucidating the complete sets of protein-protein interactions from all the possible pairs of proteins in these organisms unless we launch extremely large-scale projects or devise far more efficient and accurate experimental systems. An alternative approach is to restrict experimental targets to more likely interactions based on the information on the protein-protein interactions identified so far and on the proteins themselves. Computational prediction is a prospective approach for facilitating this laborious task. Technically, the problem of predicting interactions between two proteins can be considered as a problem of classifying the given protein pairs into two classes, interacting or non-interacting class, according to the given protein features. The score provided by the classifier is useful to assign a reliability measure to the prediction and help biologists design experimental programs. Computational prediction method can also be used for filtering likely interactions from a set of data containing a significant number of false-positives. The filtering problem is intrinsically the same as the classification problem described above. A representative example for such a dataset is the protein-protein interactions identified by high-throughput experiments such as yeast two-hybrid systems [24, 34, 35, 45, 62, 70]. Since yeast two-hybrid systems are known to yield many false positives [74], most of the previous reports often adopted scoring schemes for excluding suspicious interactions and performing convincing analyses on the obtained data. A high-performance computational prediction method facilitates this filtering process by automatically calculating a score for each protein pair. Another example of a dataset containing many false positives is the protein-protein

interactions automatically extracted from literature. A large amount of information on protein-protein interactions has been accumulated in the form of natural language texts, and computational techniques for retrieving the information have attracted considerable interests in the information extraction field [9]. Although the natural language processing techniques have been developing steadily, a certain number of false positives will always be present in the extraction results. Thus, further filtering steps are required, for example, for the automatic construction of reliable protein-protein interaction databases.

In this thesis, we propose a computational method for predicting protein-protein interactions in yeasts, mice, and humans to help researchers and experimenters reduce the time, cost, and human resources required for filtering and specifying the protein pairs that need to be examined. Our method can be applied to both 1) predicting protein-protein interactions among an input dataset created by randomly pairing the proteins of a target organism, and 2) predicting likely interactions among an input dataset containing a large number of false positives. Our method can solve both of these tasks more accurately than ever before within the same framework, but with different thresholds.

This thesis consists of four major chapters. In Chapter 2, we briefly describe the related methods for predicting protein-protein interactions and then discuss their problems. Chapter 3 describes our basic method for predicting protein-protein interactions in yeasts, mice, and humans. Section 3.2 shows how we address the problems of the previous methods by using Support Vector Machines (SVMs) and kernel methods, and Section 3.3 shows the datasets and measurements used in our study. In Section 3.4, we evaluate the performance of our SVM-based method using cross-validation analyses and show that our method outperforms the previous methods. In the same section we also show that our method can be used to filter the reliable interactions from yeast two-hybrid interactions. Predicting the interactions between all the possible pairs of proteins in a given organism (making a protein-protein interaction map) is a crucial subject in bioinformatics. The input dataset in this kind of computational prediction comprises all possible pairs of proteins in an organism, most of which are presumably non-interacting. Most of the previous methods based on supervised machine learning, including our method described in Chapter 3, are

estimated to yield a huge number of false positives for such an input dataset. In Chapter 4, we therefore propose a multiple-SVM-based method that can improve the performance in constructing hypothetical protein-protein interaction maps of yeasts and humans. Taking our SVM-based method as an example, we first show that the previous methods based on supervised machine learning yield a huge number of false positives when an input dataset contains far more non-interacting pairs of proteins than interacting pairs of proteins. Then we show that, especially on an input dataset containing a huge number of more non-interacting pairs of proteins, an approach using multiple SVMs achieves better performance than a single SVM does. This multiple-SVM-based method can also be applied to assessing the reliability of high-throughput interactions. Chapter 5 describes a Web-based service that provides a way of applying our method to predicting interactions between proteins of interest. Finally, Chapter 6 summarizes the key findings described in Chaper 3 through Chapter 5.

Chapter 2

Related works

2.1 Approaches using genomic sequences

A large amount of knowledge on genomic sequences enabled us to predict protein-protein interactions based on comparative analyses of the genomic information. In 1998, Dandekar *et al.* showed that the physical protein interactions can be predicted by analyzing the conservation of gene order [18]. An important discovery of this leading research is that one can systematically predict protein interactions without knowing nor predicting the protein structures and functions. In a subsequent year, Pellegrini *et al.* showed that the phylogenetic profile that encoded the presence or absence of genes in the genomes of several organisms can be used for assigning functions to proteins [58]. This approach is based on the assumption that proteins that function together in a cellular system tend to be preserved or eliminated simultaneously during evolution. Later, this approach was developed to predict protein-protein interactions by Date *et al.* [19]. Two proteins that are fused into one protein in a different organism are likely to interact with each other. In 1999, Enright *et al.* searched for these gene fusion events using the genomic sequences of three organisms, and found 64 fusion events involving 215 proteins [23]. Likewise, the fusion event can be considered in the domain level. Marcotte *et al.* detected 6809 putative protein-protein interactions in *Escherichia coli* and 45,502 in yeast by searching domain fusion events [47].

The main advantage of the methods described above is that they require only the information on genes in DNA sequences or on domains in protein sequences, and no other information, including protein-protein interactions, is

needed. On the other hand, a general drawback of these methods is that they yield many false positives and false negatives [71]. Huynen *et al.* compared the performances of the methods based on gene order, phylogenetic profile, and gene fusion events by applying them to the prediction of protein-protein interactions in *Mycoplasma genetalium* and found that the gene coverages were 37%, 11% and 6%, respectively [32]. The percentages of true positives were reported as 30%, 23% and 56%, respectively for the methods based on gene order, phylogenetic profile, and gene fusion events. Because of this low performance, a great deal of research interest has been shifted toward methods using other biological information, namely, the information on protein-protein interactions, which have been accumulated in recent years.

2.2 Approaches using interaction data

Most of the methods developed after 2000 used protein-protein interaction data directly. A representative method uses homology search programs such as BLAST [2]. This approach was originally introduced by Walhout *et al.* in 2000 [76] and later developed by Yu *et al.* in 2004 [81]. This method predicts an interaction between two query proteins if the homologs of these proteins are known to interact in some organisms. This method naturally assumes that interactions between proteins are conserved through evolution and the relationship is called "interologs" [76]. The reliability score of the prediction is usually calculated from the E-values of the BLAST program [81]. In the later section we will compare the performance of this method with that of our method.

Many of the conserved domains interact with each other physically and many methods based on this fact have been proposed. In 2001, Sprinzak *et al.* first demonstrated that over-represented sequence signatures (domains) can be used for predicting protein-protein interactions [66]. For domain i and j, they computed the log-odds ratio by $\log_2 F_{ij}/F_iF_j$, where F_{ij} is the observed frequency of domain pair ij in a set of interacting protein pairs, and F_i and F_j are the frequencies of domain i and j in the proteome of the organism, respectively. The log-odds ratio is positive if domain pair ij is over-represented in a set of interacting protein pairs. The log-odds ratio is calculated for every

domain pair ij and a protein pair that assigned a log-odds ratio exceeding a certain threshold is predicted to interact. This method could predict 94% of known interactions in yeast. The number of false positives is not reported in the paper. In a subsequent year, Kim *et al.* independently proposed a scoring system based on a similar idea [40]. Ng *et al.* introduced additional multipliers or weights such as the number of domains in a protein and the number of distinct experiments that identified the interaction to the method proposed by Sprinzak *et al.* [55]. In 2003, Gomez *et al.* developed the method of Sprinzak *et al.* in a different manner [25]. They assumed the domain pairs overrepresented in a set of non-interacting pairs of proteins as repulsion pairs, and reported that the resulting "attraction-repulsion" model outperformed the method of Sprinzak *et al.*

Although the domain-based methods described above have demonstrated that domain is highly informative for predicting protein-protein interactions, they have three drawbacks. First, these methods cannot distinguish a protein from other proteins that have the same domains. This implies that if protein P_1 and P_2, $P_1 \neq P_2$, contain the same domains, then these methods predict the same interaction partners for P_1 and P_2. Additional information is therefore required to distinguish P_1 from P_2. Second, these methods assume that domains are mutually independent. Suppose that, for example, protein P_1 has domain D_A and protein P_2 has domains D_B and D_C. If domain pair D_AD_B is overrepresented in a given training data set, these methods predict that P_1 interact with P_2 whether P_2 has D_C or not. However, it is apparent from an analysis of protein complexes in Protein Data Bank [8] that multiple domains take part in a physical interaction. Thus the removal of this assumption is expected to improve the prediction performance. Finally, it is difficult to modify these domain-based scoring methods so that they use biological information other than domains.

Currently, some pattern classification techniques have been applied to the prediction of protein-protein interactions. In 2001, Bock and Gough first applied a machine learning technique, SVM, to the prediction of protein-protein interactions in yeasts [10]. Their method used physicochemical properties of amino acids, i.e., charge, hydrophobicity, and surface tension, and showed good performance in separating interacting pairs of proteins with natural sequences

from non-interacting pairs of proteins with artificially shuffled sequences. However, the performance of their method should be improved to classify the pairs of natural proteins [46]. In a subsequent year, Deng *et al.* proposed a method using the Expectation Maximization algorithm [21]. Although this method was devised mainly for inferring domain-domain interactions, but it can be used to predict protein-protein interactions. The probabilities that two domains interact with each other are updated by maximizing the likelihood of observing the given protein-protein interaction data. Although this method is of theoretical interest, the performance needs to be improved for practical use.

Chapter 3

Prediction of protein-protein interactions using SVMs

3.1 Introduction

Because protein-protein interactions are key determinants of protein function, we cannot understand the cellular machinery without identifying these interactions. Although more than 16,000 protein-protein interactions in yeasts have been already identified, the total number of interactions is estimated to be much higher [3, 69, 74]. In addition, the vast majority of the interaction data have been provided by high-throughput technologies such as yeast two-hybrid systems, which are known to yield many false positives. Since *in vivo* experiments elucidating protein-protein interactions are still time-consuming and labor-intensive, methods for accurately predicting protein-protein interactions *in silico* are required to be developed.

In this chapter, we propose a computational method for predicting protein-protein interactions in yeasts, mice, and humans. The presented method uses Support Vector Machines (SVMs) to integrate different types of protein information such as protein domains, amino acid compositions, and subcellular localizations. Historically the term "interaction" has been used to indicate several relationships: one-to-one and direct relationships such as physical contact between two molecules, one-to-many and indirect relationships such as regulation of a gene expression by other gene products, and many-to-many and partially indirect relationships such as protein complexes and functional associations. Here we consider only physical contacts between two proteins.

Section 3.2 describes our method for predicting protein-protein interactions that can address the problems described in Chapter 2. Subsection 3.2.1 is a brief review of SVMs and kernel methods. The reasons why we selected SVMs as a classifier are also discussed in this subsection. Each protein pair must be expressed in term of a feature vector to be classified into the interacting or non-interacting class. Subsection 3.2.2 gives the way to construct the feature vector from the information on each protein. The protein features and data sources used are shown in Subsection 3.2.3 and 3.3.1, respectively. The performance of our SVM-based method was compared to those of two other methods: random prediction and homology-based prediction. Subsection 3.2.4 describes these prediction methods. The measurements used for assessing the performance are shown in Subsection 3.3.2. The prediction results and related discussion are shown in Section 3.4. We first give the results and discussion on the interactions between yeast proteins in Subsection 3.4.1, since most of the important parts of our classifiers were developed using the data of this organism. Finally, Subsection 3.4.2 describes the applicability of our method to the prediction of interactions between human proteins and between mouse proteins.

3.2 Methods

3.2.1 Support Vector Machines and kernel methods

Several types of information on proteins have been made available from public databases. We decided to take advantage of this information to develop a method that predicts protein-protein interactions more accurately than other methods do. Pattern classification techniques are suitable for our problem because the task of predicting protein-protein interactions can be regarded as a classification of protein pairs into the interacting or non-interacting class. We discuss the applicability of Support Vector Machines (SVMs) to our problems through testing several protein features.

Support Vector Machine (SVM) is a learning system originally developed by Vapnik *et al.* [72] for solving binary classification and regression tasks. We decided to use the SVM algorithm for the following two reasons. One is that SVMs have shown their high predictive performances in many fields including bioinformatics [10, 15]. The other is that the learning theory is well

developed and we can easily test several SVM implementations. Since the theories have been explained in many papers [12, 54, 72], here we provide only a brief description of SVMs and kernel methods.

We first consider a linear function $f : X \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ to solve binary classification problems. Suppose we have a set of labeled training examples $\{\mathbf{x}_i, y_i\}, i = 1, \ldots, l, \mathbf{x}_i \in \mathbb{R}^n, y_i \in \{-1, 1\}$. The linear function can be written as

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b, \tag{3.1}$$

where $(\mathbf{w}, b) \in \mathbb{R}^n \times \mathbb{R}$ are the parameters that control the function. The input vector $\mathbf{x}$ is assigned to the positive class if $f(\mathbf{x}) > 0$, and to the negative class if otherwise. The linear function forms, therefore, a separating hyperplane that separates positive examples from negative examples. In the linearly separable case, all training examples satisfy the following condition:

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1 \qquad \forall i. \tag{3.2}$$

Define the margin of a separating hyperplane as $2/\|\mathbf{w}\|$, which is equal to the distance between examples $\mathbf{x}_1$ and $\mathbf{x}_2$, where $\mathbf{w} \cdot \mathbf{x}_1 + b = 1$ and $\mathbf{w} \cdot \mathbf{x}_2 + b = -1$. The SVM algorithm looks for the separating hyperplane that maximizes the margin by minimizing

$$\min_{\mathbf{w}, b} \frac{1}{2}\|\mathbf{w}\|^2 \tag{3.3}$$

subject to constraints (3.2). We then form the Lagrangian by introducing positive Lagrange multiplies $\alpha_i, i = 1, \ldots, l$, which becomes

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i. \tag{3.4}$$

At the optimal point, we have

$$\frac{\partial L}{\partial b} = 0 \quad \text{and} \quad \frac{\partial L}{\partial \mathbf{w}} = 0 \tag{3.5}$$

and these can be translated into

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \quad \text{and} \quad \mathbf{w} = \sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i. \tag{3.6}$$

Substituting them into (3.4), we have a dual quadratic optimization problem:

$$\max_{\alpha} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j, \tag{3.7}$$

subject to constraints

$$\begin{aligned} \alpha_i &\geq 0, \quad \forall i \\ \sum_{i}^{l} \alpha_i y_i &= 0. \end{aligned} \tag{3.8}$$

After solving the optimization problem, all points for which $\alpha_i > 0$ holds are called support vectors. All other points have $\alpha_i = 0$.

In the linearly non-separable case, the optimal separating hyperplane can be found by introducing positive slack variables $\xi_i, i = 1, \ldots, l$ and user-adjustable parameter C, and then minimizing

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{l} \xi_i \tag{3.9}$$

subject to constraint

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1 - \xi_i \qquad \xi_i \geq 0, \ \forall i. \tag{3.10}$$

This leads to the dual quadratic problem

$$\max_{\alpha} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{3.11}$$

subject to constraints

$$\begin{aligned} 0 \leq \alpha_i &\leq C, \qquad \forall i \\ \sum_{i}^{l} \alpha_i y_i &= 0. \end{aligned} \tag{3.12}$$

In many actual problems, the linear function discussed above shows poor performance. However, the SVM algorithm can be slightly modified so that they perform still linear classification but in a different, potentially much higher dimensional space. Consider the mapping Φ, where

$$\begin{aligned} \Phi &: \mathbb{R}^n \to \mathcal{H} \\ \mathbf{x} &\mapsto \Phi(\mathbf{x}). \end{aligned} \tag{3.13}$$

Then the SVM works with the examples $\{\Phi(\mathbf{x}_i), y_i\}$, $i = 1, \ldots, l$, $y_i \in \{-1, 1\}$, $\Phi(\mathbf{x}_i) \in \mathcal{H}$. Note that the algorithm requires only inner product of two vectors. Thus if we can find a kernel function such that $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$, we need not know what the Φ is. We can then modify (3.11) as

$$\max_{\alpha} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{3.14}$$

All the discussion above still holds if we replace $\mathbf{x}_i \cdot \mathbf{x}_j$ by $K(\mathbf{x}_i, \mathbf{x}_j)$ everywhere in the algorithm. Several kernels have been proposed so far [12, 54]. We used the Gaussian radial basis function (Gaussian RBF) kernel defined by

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{y}\|^2}{\sigma^2}\right), \tag{3.15}$$

because our preliminary tests showed that this kernel outperformed the linear, sigmoidal, and polynomial kernel. For the implementation of the SVM algorithm, we used SVMTorch developed by Collobert and Bengio [16].

3.2.2 Representation of feature vector

In this subsection, we describe our procedure to construct the feature vector that addresses the problems described in Subsection 2.2.

Let $\mathbf{d} = (d_1, d_2, \ldots, d_n)^T$ be the domain vector assembled for a single protein. The element d_i is 1 if the protein has the domain i and is 0 otherwise. The order of domain is arbitrary as long as we keep it through training and testing. The dimension of the vector depends on the number n of distinct domains in all the proteins used for training. Let $\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_M$ be the vectors assembled for the same protein, where $\mathbf{f}_i$ is assembled using the ith type of protein feature (See Subsection 3.2.3 for the protein features tested). We then construct a feature vector for the protein by using

$$\mathbf{p} = \mathbf{d} \oplus w_1 \mathbf{f}_1 \oplus \cdots \oplus w_M \mathbf{f}_M, \tag{3.16}$$

where w_i denotes the weight on the ith feature with respect to the domain information and $\oplus$ denotes vector concatenation. Next, we define the feature vector for the pair of proteins 1 and 2 by

$$\mathbf{x} = \mathbf{p}_1 \oplus \mathbf{p}_2, \tag{3.17}$$

where $\mathbf{p}_1$ and $\mathbf{p}_2$ are the corresponding feature vectors calculated using Eq. (3.16).

When used with SVMs and the Gaussian RBF kernel, the feature vector defined in Eq. (3.17) can address the problems of the previous domain-based methods discussed in Subsection 2.2. The use of the protein features such as amino acid compositions makes it possible to distinguish between the proteins with the same domains. Thus our method can predict different interaction partners for proteins P_1 and P_2 even if they have the same domains. The use of the Gaussian RBF kernel defined in Eq. (3.15) avoids the assumption that domains are independent; our method can model the situation where more than two domains mediate a protein-protein interaction. Our method can predict interaction partners of a given protein more accurately than the previous domain-based methods can, because the feature vector representation permits the integration of several kinds of biological information.

3.2.3 Protein features

Our method predicts protein-protein interactions mainly based on domain information. The domain information was retrieved from the Pfam database [4] using HMMER [30]. We found 1560 domain types in yeast proteins and 2350 domain types in human proteins. We further integrated several kinds of biological information to improve the prediction performance. The four protein features tested are shown below.

The amino acid composition is known to have correlations with several biological features such as function, subcellular localization, and secondary structure fold type [56]. This feature was previously used for predicting subcellular localizations of proteins [13, 15]. We formed the amino-acid-composition vector by using

$$\mathbf{f}_A = \frac{1}{L}(N(a_1), N(a_2), \ldots N(a_{20})), \tag{3.18}$$

where the elements $N(a_1)$-$N(a_{20})$ denote the numbers of each of the standard proteinogenic amino acids and L represents the length of the protein (i. e., the total number of amino acids in the protein). The dimension of the vector is 20.

The sequential amino acid usage can be defined analogously, but this feature contains some information on the sequential order of residues in a protein:

$$\mathbf{f}_S = \frac{1}{L-1}(N(a_1a_1), N(a_1a_2), \ldots, N(a_{20}a_{20})), \tag{3.19}$$

where a_ia_j is a pair of ith and jth amino acid types in that order. The count $N(a_ia_j)$ is the number of times a_i and a_j appeared adjacently in the protein sequence. The dimension of this vector is 400.

The two amino acid indices, hydrophobicity and surface tension of amino acid solutions, were used in the previous work [10]. The use of these indices were motivated by the postulation that these hydrophobic properties of amino acid have an effect on the protein folding, hence on the interactions between proteins. These indices were obtained from the database called AAindex [39]. The dimensions of hydrophobicity and surface tension in the sequential order vary with protein length. We thus fixed the dimension of the vector to m by using a linear interpolation technique. We always used these two protein features simultaneously in the form

$$\mathbf{f}_{HT} = (h_1, h_2, \ldots, h_m, t_1, t_2, \ldots, t_m), \tag{3.20}$$

where h_i and t_i denote the interpolated values of hydrophobicity and surface tension, respectively. Preliminary we tested several values of m ranging from 50 to 400, and found that $m = 100$ worked best for our problem.

The addition of localization information is expected to improve the prediction accuracy, since a physical protein-protein interaction requires the contact between two proteins in a certain cellular location. Huh *et al.* showed that most interactions usually occur in the same cellar compartment, although there are many exceptions [31]. The localization vector can be expressed as

$$\mathbf{f}_L = (l_1, l_2, \ldots, l_m), \tag{3.21}$$

where l_i is 1 if the protein is known to localize in organelle i, and 0 otherwise. The information on protein localization was obtained from the MIPS database [50], which classifies proteins into more detailed localization categories than other databases do. We used the detailed localization information as far as possible, and the dimension of the vector is $m = 52$.

3.2.4 Other methods for comparison

Random prediction

In Section 3.4, we will examine to what extent our method performs better than a random classifier does. To this end, we estimated the measurement values for a random classifier by considering the nature of random classification; each protein pair is classified into the interacting class or into the non-interacting class randomly. Note, however, that in practical application an input dataset is expected to be skewed in the sense that most of the protein pairs in the set are no-interacting, and the performance of the random classifier depends on its preference for positive or negative result. We therefore calculated the measurement values for the random classifier by assuming that it predicts the same number of interactions as the SVM does.

Homology-based prediction

Because of its simplicity and convenience, homology-based method has been widely used for predicting protein-protein interactions [11, 76, 81]. The basic assumption is that interactions are preserved during evolution. Given a test protein pair, this method first searches for a set of homologs for each protein, and then produces a Cartesian product of the two sets. The test pair is predicted to interact if at least one pair in the Cartesian product is known to interact in some organisms. More formally, let PQ be a test protein pair, and let $S = \{P_1, P_2, \ldots, P_n\}$ and $T = \{Q_1, Q_2, \ldots, Q_m\}$ be the sets of homologs for protein P and Q, respectively. Then the pair PQ is predicted to interact if at least one pair in the set $\{P_1Q_1, P_1Q_2, \ldots, P_nQ_m\}$ is known to interact in some organism. We used BLASTP [2] for searching homologs and tested several E-values ranging from $1 \times e^{-10}$ to 10 as a cutoff threshold. Each pair in the Cartesian product was scored by

$$\text{score}(P_i, Q_j) = \max\left(E(P_i), E(Q_j)\right), \tag{3.22}$$

where $E(P_i)$ and $E(Q_j)$ represent the BLASTP E-value of proteins P and P_i and of proteins Q and Q_j, respectively. If more than one pair in the Cartesian product was matched with known interactions, we used the highest score for the test pair. We tested the performance of this homology-based method using the same protein pairs as for the SVMs.

3.3 Datasets and measurements

3.3.1 Datasets

Yeast protein-protein interactions

Physical protein-protein interactions in *S. cerevisiae* were obtained from the DIP [80] and MIPS [50] databases. Among them, we removed interactions identified by only one high-throughput project because of their low reliabilities. We defined the high-throughput method as ones used to report more than 100 interactions in a single article, following the definition of the paper [20]. This procedure yielded 4178 interactions, of which 58.2% (2430 interactions) have Pfam domains in both proteins.

Along with these interactions, we need negative data to train SVMs. We generated the negative data by compiling all possible protein pairs that were not recognized as positive in the above databases (including high-throughput results). All protein pairs that were part of a complex were removed from the negative set, since those pairs have the possibility of interacting physically with each other. This filtering yielded 20,202,318 negative candidates, of which 34.4% (6,941,526 pairs) have Pfam domains in both proteins. Although this dataset is only hypothetically negative, the probability of its containing interacting pairs is quite small. For example, the number of interactions in *S. cerevisiae* was previously estimated as 30,000 [74], which roughly corresponds to 0.17% of the total number of protein pairs (18,003,000 interactions), assuming that the number of proteins in *S. cerevisiae* is 6,000 and that each interaction has no direction. Thus, only about two in 1000 protein pairs in our yeast negative dataset actually interact.

The number of positive protein pairs is quite small compared to that of potentially negative pairs. Excessive potentially negative examples in the training set lead to yield many false negatives, and insufficient negative examples yield many false positives and lead to the fluctuation in the predictive performance. We tried some datasets changing positive/negative ratios, and finally we randomly sampled negative examples until there was four times as much negative data as there was positive data. Accordingly, 2430 positive pairs and 9720 potentially negative pairs were used for the validation. Note that the size of dataset was further decreased when additional information such as subcellular

localization was used.

Human and mouse protein-protein interactions

A set of manually curated human protein-protein interactions was obtained from HPRD [59]. After the exclusion of the high-throughput interactions as before, the dataset included 6676 physical protein-protein interactions of which 87.1% (5812 interactions) have Pfam domains in both proteins. We obtained 286 mouse protein-protein interactions from DIP, of which 80.1% (229 interactions) have Pfam domains in both proteins. Because the mouse protein-protein interaction dataset is small, we used it only for testing the applicability of the SVMs trained on pairs of human proteins to the prediction of interactions between mouse proteins.

The negative datasets were generated by randomly pairing the proteins registered in the RefSeq database [61] and removing the known interactions from them. As is the case for the prediction of yeast protein-protein interactions, four times as many negative pairs as positive pairs were used for training and testing.

3.3.2 Measurements

We used the following measurements to evaluate the performances of our SVMs. Precision (Pr) and sensitivity (Sn) were respectively defined by

$$Pr = \frac{TP}{TP + FP} \tag{3.23}$$

and

$$Sn = \frac{TP}{TP + FN}, \tag{3.24}$$

where TP, FP, FN denote the number of true positives, false positives, and false negatives, respectively. It is known, however, that there is a trade off between these two measurements. We will therefore show the F-measure (F) defined by

$$F = \frac{2 \times Pr \times Sn}{Pr + Sn}, \tag{3.25}$$

which is the harmonic mean of the precision and sensitivity. False positive rate ($FPrate$) was defined by

$$FPrate = \frac{FP}{(TN + FP)}, \tag{3.26}$$

where TN denotes the number of true negatives. The output score computed by SVMs corresponds to the distance of a test data point from the separating hyperplane in the feature space. This makes it possible to draw a receiver operating characteristic (ROC) curve, i.e. a plot of the sensitivity against the false positive rate obtained as the decision threshold is varied. In many cases, we also show an ROC score along with an ROC curve. An ROC score is calculated as the normalized area under the ROC curve. Perfect classifier has an ROC score of 1.0, while random classifier has a score of 0.5.

3.4 Results and discussion

3.4.1 Predicting yeast protein-protein interactions

Feature selection

The feature selection was performed using the yeast data for the following three reasons. First, yeast is one of the simplest model organisms whose protein-protein interactions and other biological characteristics have been extensively studied. The expected number of genes of this unicellular eukaryote is less than 6000. Second, although more than 16,000 protein-protein interactions in yeasts have been identified so far [80], the total number of interactions is estimated to be much higher [3, 74]. Thus, it is significant to predict the rest of interactions for the understanding of the interactome of the organism. Finally, the public databases such as DIP and MIPS store a large number of yeast protein-protein interactions identified by small-scale experiments as well as the ones identified by high-throughput experiments. The distinction between small-scale and high-throughput interactions is important because they have different reliabilities [74]. We used the former interactions as a reliable dataset on which we trained the SVMs and performed feature selection, and the latter as a test dataset on which we examined the filtering performance of the SVMs.

The results of the 10-fold cross validations are summarized in Table 3.1. The numbers of data and the measurement values were averaged over 10-fold cross validation tests, and the numbers in parentheses indicate the expected values of random prediction. As is the case with previous domain-based methods, the test sets might contain the protein pairs that have sequence similarity with those in the training sets. When only domain information was consid-

Table 3.1. Summary of 10-fold cross validations on yeast-protein datasets.

Feature	Number of data (Train/Test)	Dimension	Precision (Random)	Sensitivity (Random)	F-measure (Random)
D	10935/1215	3120	0.730 (0.200)	0.706 (0.193)	0.718 (0.197)
DA	10935/1215	3160	0.760 (0.200)	0.766 (0.202)	0.763 (0.201)
DS	10935/1215	3920	0.788 (0.200)	0.705 (0.179)	0.744 (0.189)
DHT	10935/1215	3520	0.758 (0.200)	0.759 (0.201)	0.758 (0.200)
DL	8780/976	3224	0.782 (0.200)	0.772 (0.198)	0.777 (0.199)
DAL	8780/976	3264	0.795 (0.200)	0.782 (0.197)	0.788 (0.198)

Several combinations of protein features are tested here: domains (D), domains and amino acid compositions (DA), domains and sequential amino acid usages (DS), domains, hydrophobicities, and surface tensions (DHT), domains and localizations (DL), and domains, amino acid compositions, and localizations (DAL). Since we used four times as many negatives as positives for training and testing, the expected results of random classification were shown in parentheses.

ered (D), the precision and sensitivity were respectively 0.730 and 0.706, and the F-measure was 0.718, 3.6 times better than random prediction. This high predictive performance indicates that the difference between interacting and non-interacting protein pairs can be well accounted for the combination of functional domains in the feature space. The additional protein features further increased the prediction accuracy. Adding the information on amino acid compositions (DA) increased the precision, sensitivity and F-measure by 0.030, 0.060 and 0.045, respectively, indicating that the information on amino acid composition is complementary to that of domain composition. The information on localizations (DL) was, as expected, shown to be effective, increasing the F-measure by 0.0590 even though the size of the data set had to be decreased. The highest F-measure, 0.788, was achieved by using both of these two additional protein features (DAL). The improvements due to the consideration of sequential amino acid usages (DS) and hydrophobicities plus surface tensions (DHT) were smaller than the improvements due to the consideration of amino acid composition or localization despite their large amount of information. One of the possible reasons is that the number of training data for these features is insufficient.

The ROC curves for the protein features that increased the F-measure by more than 4.0 percentage points are shown in Figure 3.1. For $0.200 < FPrate < 0.500$ region, the effects of additional protein features were obvious.

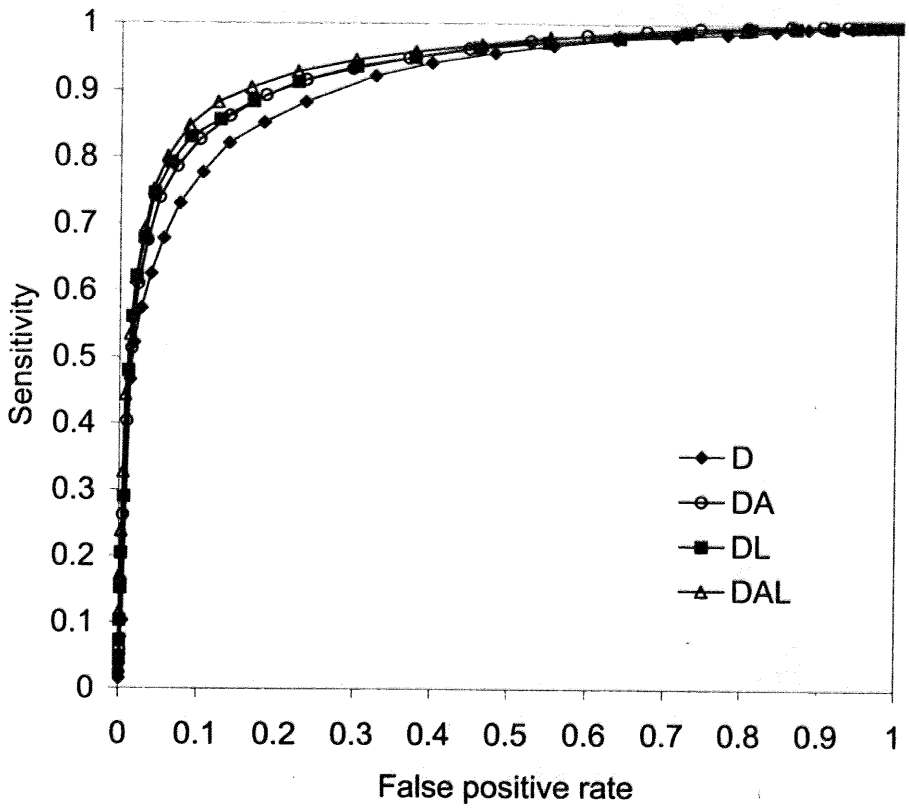

Figure 3.1. ROC curves for different protein features obtained with yeast-protein datasets. Protein features shown here are: domains (D), domains and amino acid compositions (DA), domains and localizations (DL), and domains, amino acid compositions, and localizations (DAL).

For instance, the sensitivity of D at $FPrate = 0.100$ was about 0.769, while those of DA, DL, and DAL were respectively about 0.825, 0.838, 0.859.

The calculated ROC scores of D and DAL were respectively 0.91, 0.94. The DA and DL had the same score, 0.93. Although the F-measure and the ROC score were highest for DAL, the amount of training and testing data for DAL had to be decreased. Since DL and DA had the same ROC score, DA was used in the following tests.

Yeast two-hybrid data assessment

The yeast two-hybrid system is one of the most powerful techniques to study protein-protein interactions *in vivo*. As it is a genetic method that requires only the manipulation of DNA, it can be used to reveal the large set of protein-protein interactions that are of interest. Two big projects used the high-

throughput method and revealed the huge protein network in a yeast cell [34, 35, 70]. It has also been used to infer the protein interaction map of vaccinia virus [49], *Caenorhabditis elegans* [45], *Drosophila melanogaster* [24], and humans [62]. Although the yeast two-hybrid methods are very sensitive and thus can detect transient and/or unstable physical interactions, they are known to yield many false positives. Hence, interactions detected by this method should be considered as hypotheses [57, 60, 75].

The method of the present study can be used to assess the reliability of yeast two-hybrid interactions since we excluded most of these error-prone interactions from our training data. For this test, interactions reported in the paper [34, 35, 70] were compiled, and those for which both proteins have at least one Pfam domain were extracted. After removing the redundancy from this dataset, we obtained 1979 yeast two-hybrid interactions as assessment targets.

The accuracy of the present method was tested by using as a reliable reference DIP-CORE data, which contains interactions determined by at least one small-scale experiment or by at least two high-throughput experiments, and by a homology-based prediction method [20]. As the DIP-CORE data is partially overlapped by yeast two-hybrid data, the result of prediction was evaluated by the sensitivity of this overlapping dataset, which was defined as the percentage of correct predictions of the DIP-CORE data divided by the total number of the DIP-CORE data in the yeast two-hybrid dataset. Since the number of overlapped interactions between these two datasets was small, we used the SVMs trained in the 10-fold cross validation test described above, and performed predictions 10 times after removing the interactions used as training data from the prediction targets.

The averaged numbers of predicted interactions are shown in Figure 3.2. The χ^2 test showed that there was a significant relationship between the predicted interactions and the DIP-CORE interactions ($\alpha = 0.001$). The calculated sensitivity, 0.585, was 2.3 times higher than that of random prediction. These results indicate that the present method can detect high-quality interactions among yeast two-hybrid data. On the other hand, the value of sensitivity itself seemed low when compared with the results of cross validation (see Table 3.1). One possible reason for this is that the properties of protein interactions

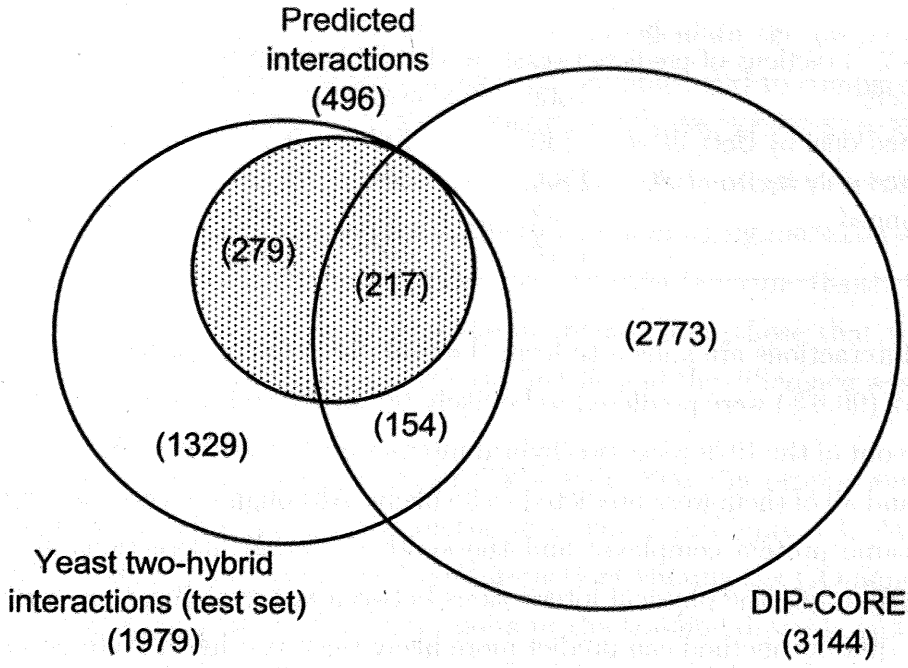

Figure 3.2. Summary of yeast two-hybrid data assessment. The test dataset used here includes the yeast two-hybrid interactions reported by Uetz *et al.* and Ito *et al.* Among 1979 yeast two-hybrid interactions, 496 interactions (25.1%) were predicted to be likely (shaded circle). Three hundred and seventy-one interactions were found in both the yeast two-hybrid and DIP-CORE dataset, of which 217 interactions (58.5%) were predicted to be likely. Note that the numbers of interactions used for this test were fewer than the numbers of interactions originally reported because our method requires a protein to have at least one domain.

seem to differ between the yeast two-hybrid data and the training/testing data used for the cross validation. For example, when focusing on the interactions between single domain proteins, the proportion of the testing protein pairs containing the single-domain protein pairs that also appeared in the training set was 20.3% on average, whereas this proportion in the yeast two-hybrid test was only 11.6%. The difference between these proportions is significant ($\alpha = 0.001$).

In Table 3.2, we illustrate the fractions of the predicted interactions in different datasets. The yeast two-hybrid interactions reported by Uetz *et al.* were more likely to be predicted than that reported by Ito *et al.* were. Nearly half of the interactions identified by both projects were predicted, indicating that these interactions are more likely to be true interactions.

One hundred and seventy-four pairs of proteins out of the 1979 yeast two-

Table 3.2. Fractions of predicted yeast two-hybrid interactions.

Data sets	Total	Predicted	Fraction
Reported only by Uetz *et al.*	330	112	0.339
Reported only by Ito *et al.*	1566	344	0.220
Overlapped	83	40	0.482
Total	1979	496	0.251

hybrid interactions are known to form the same protein complexes. Of which, 167 pairs (96.0%) were predicted to be likely by the SVM. Likewise, 10 pairs of proteins out of the 1979 yeast two-hybrid interactions are known to be synthetic lethal, and all of them were predicted to be likely. Although the co-membership of the same protein complexes and the synthetic lethal interactions do not necessarily imply the physical interactions between two proteins, these results suggest that our method can predict more likely yeast two-hybrid interactions.

In summary, the present method can detect high-quality interactions in the yeast two-hybrid data, but the number of these interactions may be underestimated because yeast two-hybrid data contains many domain-domain pairs that do not appear in the training dataset.

Predicting unknown interactions

We next predicted the putative interactions in a dataset created by randomly pairing the yeast proteins. Since some of the pairs were used as "negative" data in the training dataset, we repeated the training and testing procedure three times with varying the "negative" sets, and the pairs that were predicted in all trials were extracted as reliable data.

The biological relevance of the predicted interactions was evaluated by comparing the depth 4 GO annotations in the "biological process" hierarchy. Here the depths 0 and 1 were respectively defined as "Gene_Ontology" and "biological_process". According to this definition, the depth 4 annotations are "carbohydrate metabolism," "alcohol metabolism," "oxidative phosphorylation," "response to oxidative stress," etc. In this test, protein pairs were discarded when at least one of the proteins was not annotated or was assigned a GO annotation with higher than depth 4. When the protein was assigned a GO annotation with lower than depth 4, the upper GO annotation with depth 4 was re-assigned. The semantic difference between "Is-a" and "Part-of" was

ignored since the main objective here is to calculate the proportion of predicted interactions having any biological relevance and to compare them with that of "negative" pairs. When at least one protein of the pair had multiple GO annotations re-assigned, all combinations of them were compared. When this evaluation method was used, the rate of interacting pairs (those that were used as positive in the training data set) sharing the same depth 4 GO annotation was 92.7%, while that of the unknown pairs (those that were used as negative in the training data set) was 56.0%, and the difference was significant ($\alpha = 0.001$).

The relationship between the SVM score and the proportion of protein pairs sharing the same GO annotation is shown in Figure 3.3. At each SVM score, the proportion of predicted interactions sharing the GO annotation was far higher than that of negative pairs in the training dataset, and this difference increased as the decision threshold became higher. Homo-dimers tend to have high scores, and always share the GO annotation. For this reason, homo-dimers were excluded from the predicted interactions and the proportions were recalculated, but the exclusion made little difference. This evaluation is limited, since sharing the same GO annotations does not necessarily imply physical interaction between two proteins, and vice versa. The strong relationship between the proportion and the SVM score indicates, however, the present method can detect more likely interactions among the unknown protein pairs. The number of predicted interactions is also shown in Figure 3.3 on a logarithmic scale. The number of interactions which had the scores of more than 0, 0.5, 1.0, 1.5, and 2.0 were respectively 173876, 101217, 24250, 3703, 503 and 59. Several rough estimations of the total number of interactions in yeasts have been made without definite evidences [3, 69, 74], and they range from 8000 to 30000. Compared with these numbers, those predicted by the present method seems to indicate over-prediction of interactions in some cases. One possible reason for this is that the present method cannot distinguish very similar proteins in paralog pathways, such as serine/threonine protein kinases. For instance, FUS3 and HOG1, which are involved in the pheromone-induced signal pathway and in the maintenance of water homeostasis, respectively, are paralog serine/threonine kinases. It is known that the STE7 serine/threonine kinase phosphorylates the FUS3 in the pheromone-induced signal pathway,

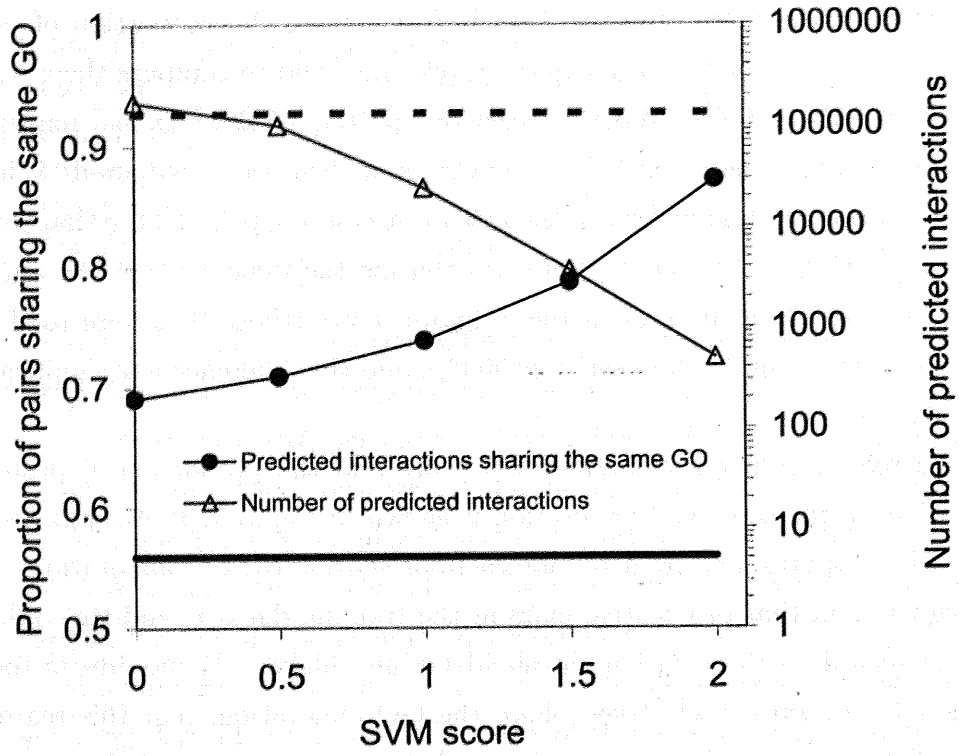

Figure 3.3. Fractions of protein pairs sharing the same GO annotations. The fraction of the predicted interactions sharing the same GO annotations is plotted against the SVM-score threshold used for the prediction (solid line with closed circles). The number of interactions predicted at a given threshold (solid line with open triangles) is shown on the right-hand axis on a logarithmic scale. For reference, the fractions of the positive and negative examples in the training dataset sharing the same GO annotations are shown as horizontal dashed line and solid line, respectively.

and this leads to the erroneous prediction of interaction between STE7 and HOG1. These paralog pathway cases can be addressed in some extent by using the phylogenetic tree of interaction proteins [41]. To filter the inevitable errors that occur when the present method is applied, it needs to be combined with another screening method. Nevertheless, it can help biologists gain insights by identifying the interactions that are likely to occur in the cellular process.

The top 20 predicted interactions are listed in Table 3.3. Of these, three interactions were confirmed by references (YPD of Incyte, http://www.incyte.com/control/home) and PRIME [42]. In addition, many of them seem to have functional relationships even if they do not have the same GO annotation. For example, CMD1 is a calmodulin, a calcium-binding protein, and is involved in

many processes including cell polarization, nuclear division, and chromosome maintenance. NSP1, on the other hand, is localized in nuclear membranes and nuclear pores and is involved in nucleocytoplasmic transport. Its depletion experiment also suggests the role of NSP1 in nuclear division process [53]. There is a logical interaction between CMD1 and NSP1 in nuclear division process.

All predicted interactions between yeast proteins are available from http://cb.k.u-tokyo.ac.jp/~dohkan/.

Comparison with other methods

We first compared the performance of the SVM trained using the information on domains and amino acid compositions with that of a homology-based method. Figure 3.4 shows the ROC curves for both methods. Except for $FPrate < 0.020$ region, our SVM clearly outperformed the homology-based method. For instance, the sensitivity of the SVM at $FPrate = 0.150$ was 0.850, while that of the homology-based method was 0.638. This result indicates that our method is especially useful for a first screening of likely interactions from numerous protein pairs. To investigate to what extent our SVMs can predict protein-protein interactions that cannot be predicted by the homology-based method, we removed all the interactions predicted by the homology-based method from the test datasets used in the previous 10-fold cross validation and recalculated an averaged F-measure on the datasets. This test was performed repeatedly using different E-value thresholds for the homology-based method. As shown in Fig. 3.5, our SVM achieved far better results than random prediction. This implies that, as shown in Fig. 3.4, although there is little difference in sensitivity at very low false-positive-rate region, our SVM can predict protein-protein interactions that cannot be predicted by the homology methods.

Several other methods for predicting yeast protein-protein interactions have been proposed (see Section 2). Bock and Gough used SVMs and physicochemical properties of amino acid residues such as hydrophobicity and surface tension [10]. We evaluated the contribution of these features to the prediction by retraining our SVMs without using the domain information and then testing. As a result, the precision, sensitivity, and F-measure were respectively decreased

Table 3.3. Top 20 of predicted interactions.

Predicted pair	GO ID	GO term
CYC8	GO:0016481	negative regulation of transcription
CYC8	GO:0016481	negative regulation of transcription
CMD1	GO:0007010	cytoskeleton organization and biogenesis
	GO:0007067	mitosis
	GO:0007114	budding
NUP116	GO:0006388	tRNA splicing
	GO:0006406	mRNA-nucleus export
	GO:0006407	rRNA-nucleus export
	GO:0006408	snRNA-nucleus export
	GO:0006409	tRNA-nucleus export
	GO:0006606	protein-nucleus import
	GO:0006607	NLS-bearing substrate-nucleus import
	GO:0006608	snRNP protein-nucleus import
	GO:0006609	mRNA-binding (hnRNP) protein-nucleus import
	GO:0006610	ribosomal protein-nucleus import
	GO:0006611	protein-nucleus export
	GO:0006999	nuclear pore organization and biogenesis
CMD1	GO:0007010	cytoskeleton organization and biogenesis
	GO:0007067	mitosis
	GO:0007114	budding
NUP100	GO:0006406	mRNA-nucleus export
	GO:0006407	rRNA-nucleus export
	GO:0006408	snRNA-nucleus export
	GO:0006409	tRNA-nucleus export
	GO:0006607	NLS-bearing substrate-nucleus import
	GO:0006608	snRNP protein-nucleus import
	GO:0006609	mRNA-binding (hnRNP) protein-nucleus import
	GO:0006610	ribosomal protein-nucleus import
	GO:0006611	protein-nucleus export
	GO:0006999	nuclear pore organization and biogenesis
CYC8	GO:0016481	negative regulation of transcription
MCM1*	GO:0006270	DNA replication initiation
	GO:0006357	regulation of transcription from Pol II promoter
PAN1	GO:0000147	actin cortical patch assembly
	GO:0000910	cytokinesis
	GO:0006897	endocytosis
	GO:0007120	axial budding
	GO:0007121	polar budding
PAN1	GO:0000147	actin cortical patch assembly
	GO:0000910	cytokinesis
	GO:0006897	endocytosis
	GO:0007120	axial budding
	GO:0007121	polar budding
STE20	GO:0000282	bud site selection
	GO:0000750	signal transduction during conjugation with cellular fusion
	GO:0006468	protein amino acid phosphorylation
	GO:0007124	pseudohyphal growth
STE20	GO:0000282	bud site selection
	GO:0000750	signal transduction during conjugation with cellular fusion
	GO:0006468	protein amino acid phosphorylation
	GO:0007124	pseudohyphal growth
CLA4	GO:0000283	establishment of cell polarity (sensu Saccharomyces)
	GO:0000910	cytokinesis
	GO:0006468	protein amino acid phosphorylation
	GO:0007118	apical bud growth
	GO:0007266	Rho protein signal transduction
CLA4	GO:0000283	establishment of cell polarity (sensu Saccharomyces)
	GO:0000910	cytokinesis
	GO:0006468	protein amino acid phosphorylation
	GO:0007118	apical bud growth
	GO:0007266	Rho protein signal transduction
SMD3	GO:0000398	nuclear mRNA splicing, via spliceosome
SMD3	GO:0000398	nuclear mRNA splicing, via spliceosome
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
RPT5	GO:0006511	ubiquitin-dependent protein catabolism
MRS11	GO:0006628	mitochondrial translocation
MRS11	GO:0006628	mitochondrial translocation
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
HSP104	GO:0006457	protein folding
	GO:0006950	response to stress
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
RPT6	GO:0006511	ubiquitin-dependent protein catabolism

Predicted pair	GO ID	GO term
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
RPT1	GO:0006511	ubiquitin-dependent protein catabolism
CMD1	GO:0007010	cytoskeleton organization and biogenesis
	GO:0007067	mitosis
	GO:0007114	budding
NSP1	GO:0000055	ribosomal large subunit-nucleus export
	GO:0006405	RNA-nucleus export
	GO:0006406	mRNA-nucleus export
	GO:0006407	rRNA-nucleus export
	GO:0006408	snRNA-nucleus export
	GO:0006409	tRNA-nucleus export
	GO:0006606	protein-nucleus import
	GO:0006607	NLS-bearing substrate-nucleus import
	GO:0006608	snRNP protein-nucleus import
	GO:0006609	mRNA-binding (hnRNP) protein-nucleus import
	GO:0006610	ribosomal protein-nucleus import
	GO:0006611	protein-nucleus export
	GO:0006999	nuclear pore organization and biogenesis
SMX2	GO:0000398	"nuclear mRNA splicing, via spliceosome"
SMX2†	GO:0000398	"nuclear mRNA splicing, via spliceosome"
RVS167	GO:0006897	endocytosis
	GO:0006970	response to osmotic stress
	GO:0007121	polar budding
SPT15	GO:0006360	transcription from Pol I promoter
	GO:0006367	transcription initiation from Pol II promoter
	GO:0006384	transcription initiation from Pol III promoter
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
RPT3	GO:0006511	ubiquitin-dependent protein catabolism
CMD1	GO:0007010	cytoskeleton organization and biogenesis
	GO:0007067	mitosis
	GO:0007114	budding
NUP145	GO:0006388	tRNA splicing
	GO:0006406	mRNA-nucleus export
	GO:0006407	rRNA-nucleus export
	GO:0006408	snRNA-nucleus export
	GO:0006409	tRNA-nucleus export
	GO:0006606	protein-nucleus import
	GO:0006607	NLS-bearing substrate-nucleus import
	GO:0006608	snRNP protein-nucleus import
	GO:0006609	mRNA-binding (hnRNP) protein-nucleus import
	GO:0006610	ribosomal protein-nucleus import
	GO:0006611	protein-nucleus export
	GO:0006999	nuclear pore organization and biogenesis
MAK31	GO:0006474	N-terminal protein amino acid acetylation
	GO:0019048	virus-host interaction
MAK31	GO:0006474	N-terminal protein amino acid acetylation
	GO:0019048	virus-host interaction
CDC48	GO:0006511	ubiquitin-dependent protein catabolism
	GO:0006906	nonselective vesicle fusion
	GO:0006915	apoptosis
	GO:0007049	cell cycle
	GO:0015031	protein transport
	GO:0030433	ER-associated protein catabolism
RFC1	GO:0006272	leading strand elongation
	GO:0006281	DNA repair
	GO:0006298	mismatch repair

* Confirmed by PRIME [42]

† Confirmed by YPD (http://www.incyte.com/control/home)

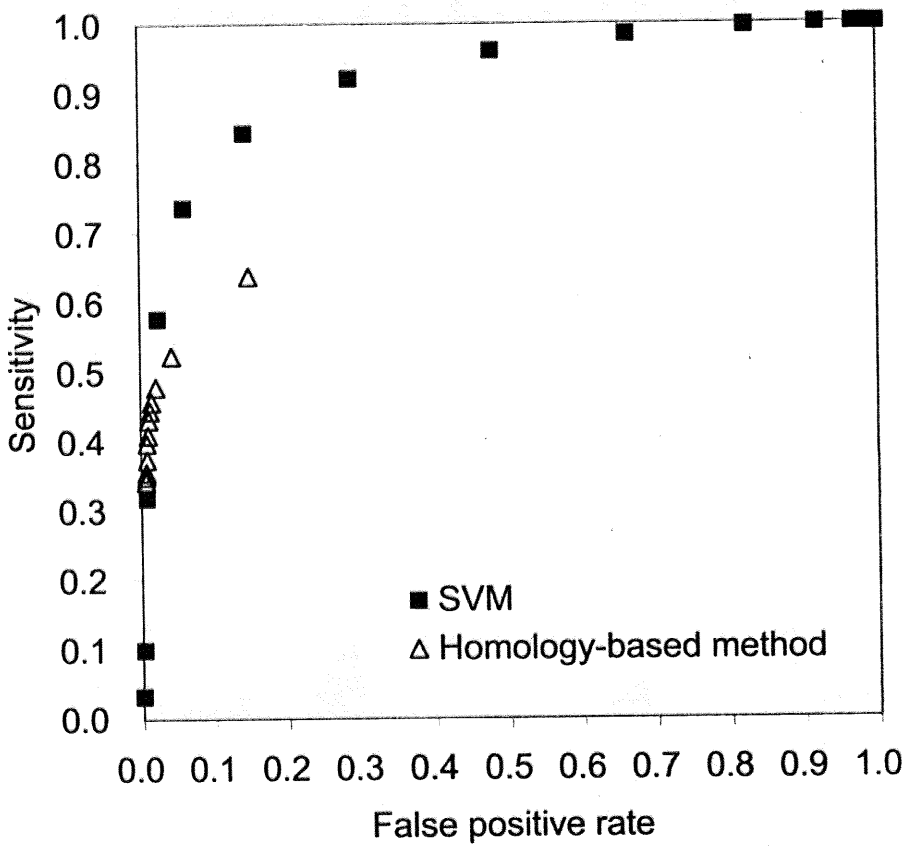

Figure 3.4. ROC curves for SVM and homology-based method applied to yeast-protein datasets. As before, we trained the SVM using the information on domains and amino acid compositions of proteins.

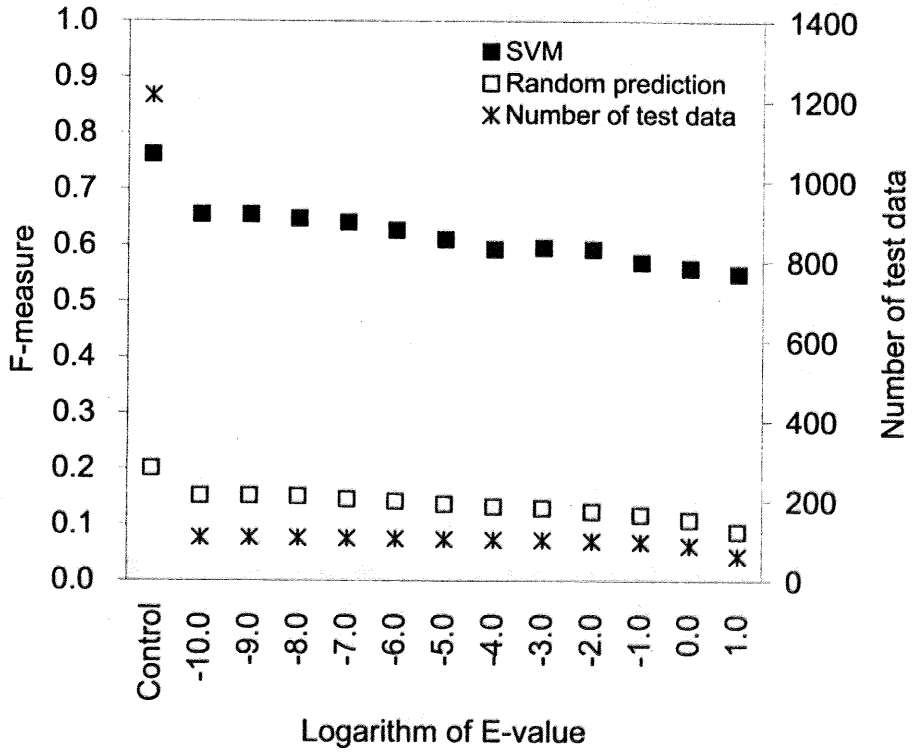

Figure 3.5. Performance of SVM obtained with the yeast-protein test datasets with all the interactions predicted by the homology-based method removed. The x-axis indicates the logarithms of E-value thresholds used for the homology-based method. The values of "Control" indicate the results of the usual 10-fold cross validation shown in Table 3.1.

to 0.713, 0.649, and 0.679 in a 10-fold cross validation test (cf. Table 3.1). This result implies that only the use of these physicochemical properties is not sufficient to predict interactions accurately, and domain information is quite informative.

Our method is novel in that it can take the effect of multiple domains in account. The maximum-likelihood estimation-based method proposed by Deng *et al.* assumed the domain independence [21] and achieved a sensitivity of 0.797 and a precision of 0.390. The attraction-repulsion model proposed by Gomez *et al.* chose the most probable domain-domain interaction to score a protein pair and achieved an ROC score of about 0.82 [25] (our result: ROC score = 0.94 with features DAL). These methods did not model the case where more than two domains mediate a protein-protein interaction, and deciding which domain-domain interaction is the key determinant of the protein-protein interaction is problematic. Han *et al.* recently introduced the ideas of domain combination and domain combination pairs and achieved a sensitivity of 0.86 and a specificity of 0.56 [26] (our results: sensitivity = 0.78 and specificity = 0.95 with features DAL). Although this method considers the effects of multiple domains, protein-protein interactions are not always determined by only domain compositions. In this context, our method is advantageous in that the effects of multiple domains and other protein features can be considered by combining them into a feature vector. Although the evaluation of the method itself is quite difficult due to the fact that the performance of machine learning is largely influenced by validation datasets, our method yielded relatively high prediction accuracies compared to those reported previously.

3.4.2 Predicting mammalian protein-protein interactions

Predicting human protein-protein interactions

Predicting mammalian protein-protein interactions is a challenging problem on which few previous works focused, and is far more difficult than predicting yeast protein-protein interactions because the number of mammalian proteins and of all the possible pairs of them are much larger than those of yeast proteins. In this subsection, we assess the applicability of our method to predicting human protein-protein interactions based on cross-validation. Because subcellular localizations of many human proteins are still unknown, we trained SVMs using the information on only domains and amino acid compositions.

We list the averaged results of 10-fold cross validations in Table 3.4. Here we tested two sets of domain information: a set of all domains observed in the human proteins (D_H) and a set of all domains observed in yeast proteins (D_Y), each with and without amino acid compositions (D_HA and D_YA, respectively). The motivation of using the domains observed in yeast proteins is to reduce the dimension size of the feature vector and to compare the performance of the SVMs with that of the SVMs trained on the yeast protein pairs. The results listed in Table 3.4 indicates that the information on amino acid compositions is effective for predicting human protein-protein interactions as well. A comparison of D_H and D_Y and of D_HA and D_YA indicates that the SVMs trained using domains observed in human proteins outperform the SVMs trained using domains within yeast proteins. The highest F-measure, 0.776, was achieved by using the feature D_HA and was close to the result of the SVM trained on the yeast protein pairs using the information on domains and amino acid compositions (F-measure: 0.769, see Table 3.1). The SVM trained on the human protein-protein interactions using the feature D_Y had worse results than the SVM trained on the yeast protein-protein interactions using the feature D, although the number of protein pairs used for the training and testing was greater in the former and the dimensions of the feature vectors were the same between these SVMs (see Table 3.1). Since the former SVM did not use the information on domains that were in the human proteins but not in the yeast proteins, this result implies that emergence of new domains in a protein may alter the protein's interaction partners.

Table 3.4. Summary of 10-fold cross validations on human-protein datasets.

Feature	Number of data (Train/Test)	Dimension	Precision (Random)	Sensitivity (Random)	F-measure (Random)
D_Y	14328/1592	3120	0.765 (0.200)	0.624 (0.163)	0.687 (0.180)
$D_Y A$	14328/1592	3460	0.774 (0.200)	0.716 (0.185)	0.744 (0.192)
D_H	26154/2906	4700	0.798 (0.200)	0.681 (0.171)	0.734 (0.184)
$D_H A$	26154/2906	4740	0.797 (0.200)	0.757 (0.190)	0.776 (0.195)

We tested two sets of protein domains: a set of domains in yeast proteins (D_Y) and a set of domains in human proteins (D_H), each with and without the information of amino acid compositions (A). As in Table 3.1, the numbers in parentheses indicates the expected performances of random classification on the datasets containing four times as many negatives as positives.

We compared the performance of SVM trained on human protein pairs with that of SVM trained on yeast protein pairs using ROC curves (Fig.3.6). The protein features used here were all domains observed in each of the organism's proteins and amino acid compositions. As indicated, there was no clear difference between the two curves. In fact, the ROC score, 0.93, was the same for both SVMs. Together with the results listed in Table 3.4, this result implies that our method can be applied to the prediction of interactions between human proteins.

We next compared the performance of the SVM trained on human protein pairs with that of a homology-based method in predicting human protein-protein interactions. Figs. 3.7 and 3.8 are the human-protein results corresponding to the yeast-protein results shown in Figs. 3.4 and 3.5. As expected from the results on the yeast-protein datasets, the SVM trained on the human protein pairs clearly outperformed the homology-based prediction.

To investigate to what extent the performance of SVM is influenced by the number of training data, we randomly removed a certain proportion of protein pairs from the yeast-protein and human-protein datasets and retrained SVMs on those datasets. As shown in Fig. 3.9, the values of F-measure decreased as we decreased the number of data used for training. From this figure, it can be roughly estimated that if the number of protein-protein interactions is doubled, corresponding value of F-measure will be close to 0.85.

Currently, few machine-learning-based methods have been proposed to predict protein-protein interactions in mammals. Recently, Martin *et al.* used a

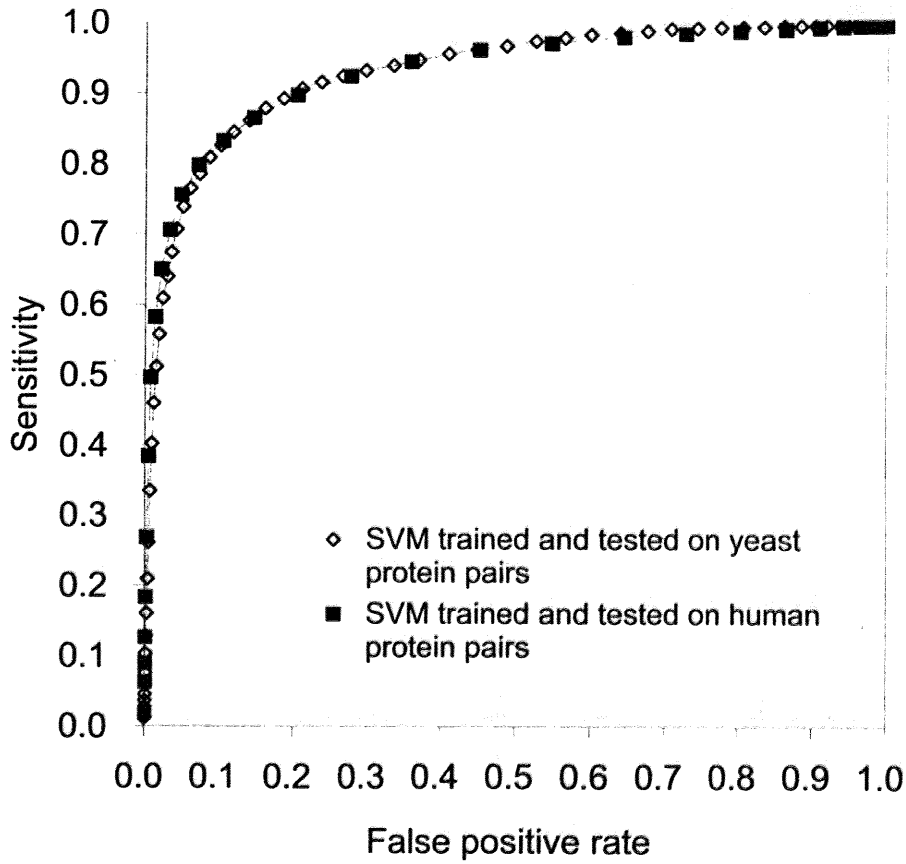

Figure 3.6. Comparison of ROC curves between SVM trained and tested on yeast protein pairs and SVM trained and tested on human protein pairs. The protein features used are all domains within each organism and amino acid compositions of proteins.

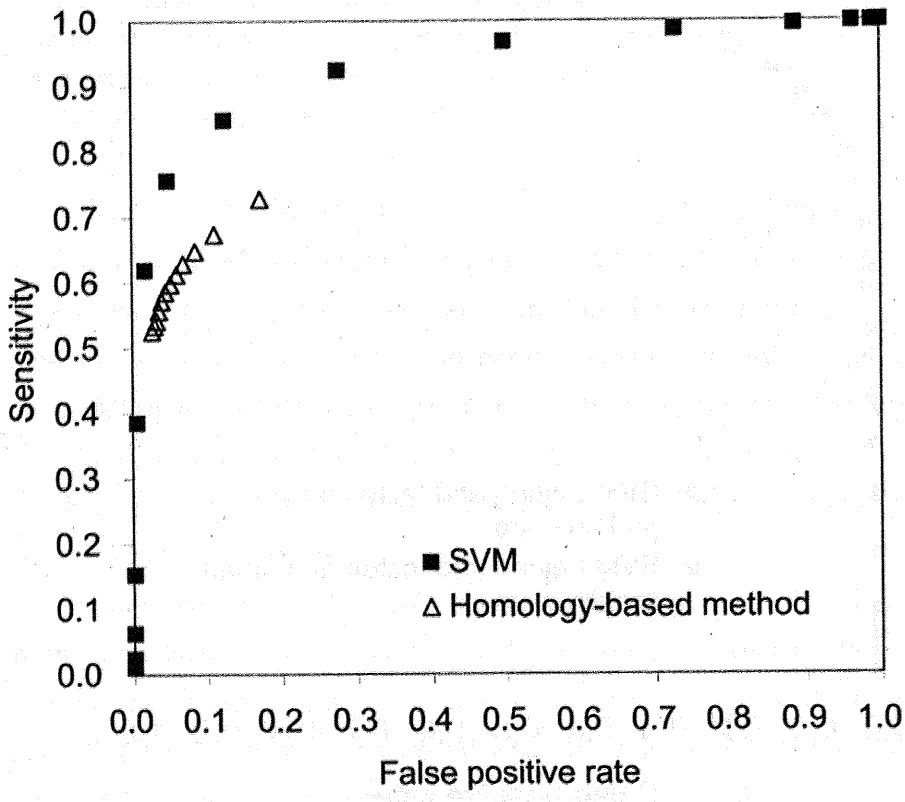

Figure 3.7. ROC curves for SVM and homology-based method applied to human-protein datasets. ROC curves shown here are the human-protein results corresponding to the yeast-protein results shown in Fig 3.4. We trained SVM using the information on all domains within human proteins and amino acid compositions.

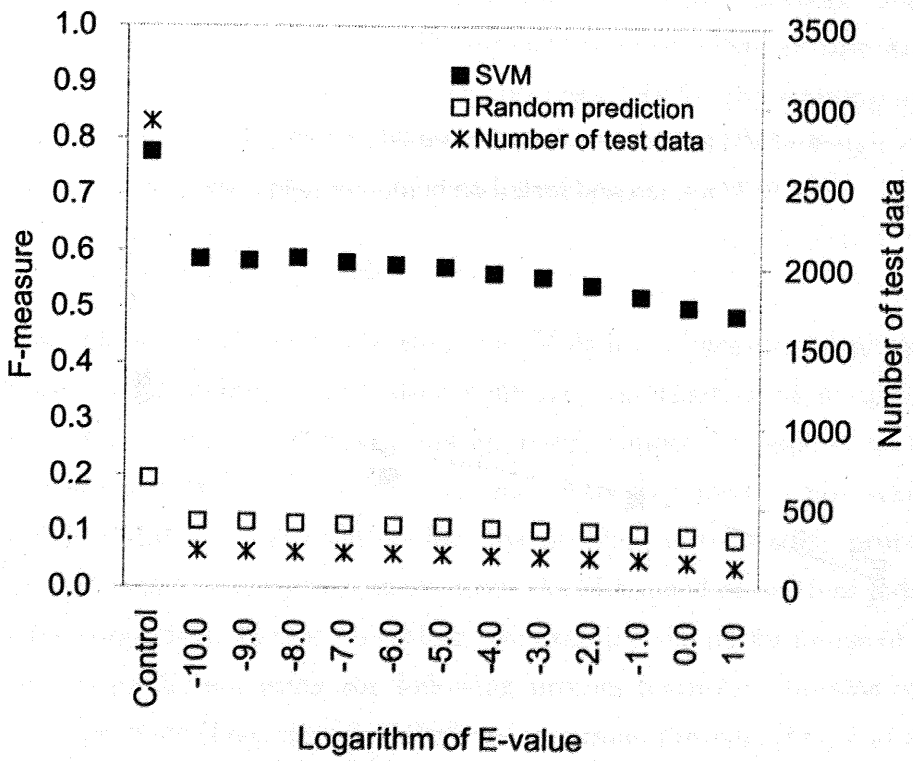

Figure 3.8. Performance of SVM obtained with the human-protein test datasets with all the interactions predicted by the homology-based method removed. This human-protein result corresponds to the yeast-protein result shown in Fig. 3.5. As in Fig. 3.5, the x-axis indicates the logarithms of E-value thresholds used for the homology-based method, and the values of "Control" indicate the results of the usual 10-fold cross validation shown in Table 3.4.

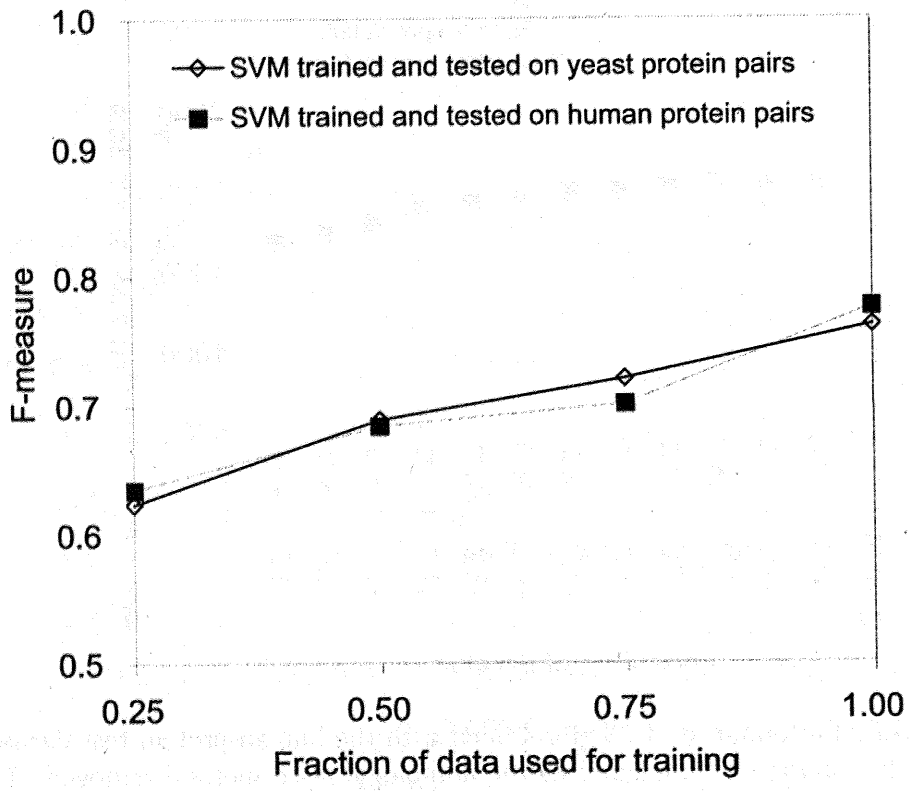

Figure 3.9. Relationship between the fraction of training data used for training SVM and corresponding F-measure.

protein feature similar to our sequential amino acid usage [48] with SVM and predicted protein-protein interactions in yeasts and humans. The precision and sensitivity were reported to be 0.715 and 0.632 in predicting interactions in yeasts, and 0.722 and 0.662 in predicting interactions in humans (our method: 0.760 precision and 0.766 sensitivity in predicting interactions in yeasts and 0.797 precision and 0.757 sensitivity in predicting interactions in humans, using domains and amino acid compositions as protein features). Note that our prediction problem was more difficult than their problem because whereas they used the same number of positives and negatives for the training and testing, we used four times as many negatives as positives. This rough comparison implies that our method is superior to their method.

Cross-species prediction

Heretofore, we trained and tested our SVMs in an intra-organism manner: the datasets for training and testing contained information about proteins from the same species. However, this approach cannot be applied to the organism whose protein-protein interactions are rarely known or not available from public databases. A solution for this problem is to predict protein-protein interactions of a target organism using SVM trained on protein pairs of a different organism. In this subsection, we examine the performance of this cross-species prediction using the following protein features: domains observed in yeast proteins (D_Y), domains observed in human proteins (D_H) and amino acid compositions of proteins (A).

The performances of SVMs trained on yeast protein pairs in predicting human protein-protein interactions were shown in Table 3.5. The values of precision, sensitivity, and F-measure obtained using the protein feature D_Y were 0.494, 0.417, and 0.452, respectively, indicating that the set of yeast protein-protein interactions does not contain enough information to predict human protein-protein interactions. The performance was further decreased by adding the information on amino acid compositions. Together with the fact that the test set contains only the pairs of human proteins retaining the domains observed in yeast proteins, these prediction results imply that proteins have changed their interaction partners during evolution. Conversely, we predicted yeast protein-protein interactions using SVMs trained on human

Table 3.5. Performance of SVM trained on yeast protein pairs in predicting human protein-protein interactions.

Feature	Number of data (Train/Test)	Dimension	Precision (Random)	Sensitivity (Random)	F-measure (Random)
D_Y	12150/15920	3120	0.494 (0.200)	0.417 (0.169)	0.452 (0.183)
$D_Y A$	12150/15920	3160	0.394 (0.200)	0.474 (0.240)	0.431 (0.218)

The protein features tested are domains within yeast proteins (D_Y) and amino acid compositions of proteins (A).

Table 3.6. Performance of SVM trained on human protein pairs in predicting yeast protein-protein interactions.

Feature	Number of data (Train/Test)	Dimension	Precision (Random)	Sensitivity (Random)	F-measure (Random)
D_Y	15920/12150	3120	0.573 (0.200)	0.284 (0.099)	0.380 (0.133)
$D_Y A$	15920/12150	3160	0.503 (0.200)	0.429 (0.170)	0.463 (0.184)

The protein features tested are domains within yeast proteins (D_Y) and amino acid compositions of proteins (A).

protein pairs and found that the performances of SVMs were far worse than that of the same SVMs in predicting human protein-protein interactions (Table 3.6). Here the addition of information on amino acid compositions increased the performance of SVM. We have no hypothesis to explain this phenomenon. We next predicted mouse protein-protein interactions using SVMs trained on human protein pairs. As shown in Table 3.7, the performance of SVMs was similar to that of the same SVMs in predicting human protein-protein interactions. The highest F-measure of 0.765 was obtained using the protein features D_H and A, indicating that the SVM trained on human protein pairs can be applied to the prediction of mouse protein-protein interactions.

In summary, the performance of cross-species prediction depends on the evolutionary distance between the source and target organism. Prediction of protein-protein interactions in, for example, chimpanzee, rat, and guinea pig may be possible by using the SVM trained on the human protein pairs.

Table 3.7. Performance of SVM trained on human protein pairs in predicting mouse protein-protein interactions.

Feature	Number of data (Train/Test)	Dimension	Precision (Random)	Sensitivity (Random)	F-measure (Random)
D_Y	15920/660	3120	0.733 (0.200)	0.500 (0.136)	0.595 (0.162)
$D_Y A$	15920/660	3160	0.742 (0.200)	0.720 (0.194)	0.731 (0.197)
D_H	26154/1145	4700	0.720 (0.200)	0.545 (0.152)	0.621 (0.172)
$D_H A$	26154/1145	4740	0.807 (0.200)	0.727 (0.180)	0.765 (0.190)

The protein features tested are domains within yeast proteins (D_Y), domains within human proteins (D_H), and amino acid compositions of proteins (A).