Chapter 4

Improving the performance of SVMs for constructing hypothetical protein-protein interaction maps

4.1 Introduction

A computational method for predicting a comprehensive set of protein-protein interactions, for making a protein-protein interaction map, from sequence information can help biologists reduce the time, cost, and human resources required for filtering and specifying the protein pairs that need to be examined. 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. Many of the methods developed after 2000 for predicting protein-protein interactions are based on supervised machine learning techniques, such as random decision forests [14] and Support Vector Machines (SVMs) [5, 10, 22, 48], and perform well when the input dataset contains the same number of interacting and non-interacting pairs of proteins. When the input dataset contains far more non-interacting pairs than interacting pairs, however, these methods yield a large number of false positives [48]. The objective of this chapter is to evaluate and improve the performance of a method based on supervised machine learning when it is used to construct a yeast protein-protein interaction map and a human protein-protein interaction map.

Most of the previous studies using methods based on supervised machine

learning to predict protein-protein interactions have used training datasets containing approximately the same numbers of interacting and non-interacting pairs of proteins. Such a training dataset is inadequate, however, because in an actual cell the number of non-interacting pairs (negatives) is much larger than that of interacting pairs (positives), and a classifier trained on a dataset with a small fraction of non-interacting pairs will yield many false positives. Using our previously developed SVMs to predict protein-protein interactions, we show in Subsection 4.4.1 and 4.4.2 that an SVM trained on data containing the same number of positives and negatives does not perform as well as one trained on data containing more negatives than positives. We then report in Subsection 4.4.3 that an approach using multiple SVMs can predict more of the likely interactions in a test dataset created by randomly pairing proteins than a single SVM can. Finally in Subsection 4.4.4 we demonstrate that our method can also be used to extract likely interactions from high-throughput interactions, which is another important problem in obtaining reliable interaction maps.

4.2 Methods

4.2.1 SVM-based prediction

As in Chapter 3, we train SVMs using the information on domains and amino acid compositions of proteins. Briefly, let $\mathbf{d} = (d_1, d_2, \ldots, d_n)$ 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 dimension of the vector depends on the number n of distinct domains in all the proteins used for training. Let $\mathbf{f} = \big(N(a_1), N(a_2), \ldots, N(a_{20})\big)/L$ be the amino-acid-composition vector assembled for the same protein. 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). We then assemble a feature vector for the protein by using

$$\mathbf{p} = \mathbf{d} \oplus w\mathbf{f}, \tag{4.1}$$

where $\oplus$ denotes vector concatenation and w denotes the weight of the amino-acid-composition information relative to the domain information. The feature

vector for the pair of proteins 1 and 2 can then be written as

$$\mathbf{x} = \mathbf{p}_1 \oplus \mathbf{p}_2, \tag{4.2}$$

where $\mathbf{p}_1$ and $\mathbf{p}_2$ are the corresponding feature vectors calculated using Eq. (4.1).

Given a training dataset that includes both interacting and non-interacting pairs of proteins, SVMs look for a decision boundary that maximizes the margin between the two sets of data points. Because SVMs can perform binary classifications in potentially higher-dimensional spaces by using the kernel method, we used the Gaussian radial basis function (RBF) kernel defined by

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{y}\|^2}{\sigma^2}\right), \tag{4.3}$$

where $\mathbf{x}$ and $\mathbf{y}$ are feature vectors calculated using Eq. (4.1) and σ is a parameter. As described in Chapter 3, the advantage of using this kernel for our problem is that it eliminates the need to assume domain independence: the effect of multiple domains in a protein on a protein-protein interaction can be captured using the Gaussian RBF kernel. The weight w and parameter o were empirically determined as $w = 10$ and $\sigma = 5$. In a test phase an SVM calculates a score, which is the distance in the feature space from the data point to the nearest point on the decision boundary. When the default threshold is 0, a pair of proteins assigned a positive score is predicted to interact and a pair assigned a negative score is predicted not to interact. The threshold can be adjusted so that an SVM yields fewer false positives; this corresponds to a parallel shift of the decision boundary. In this chapter, 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.

4.2.2 Prediction using multiple SVMs

We think that a training dataset containing the same number of positives and negatives causes a classifier trained on it to yield many false positives when predicting interactions and thus that the false positive rate can be reduced by increasing the number of negatives in the training dataset. To test this hypothesis, we used training datasets containing more negatives than positives.

The number of negatives in a training dataset has a practical limit, however, because the training time increases significantly as we increase the number of negatives used for training. We tackled this problem by using multiple SVMs, each trained on a dataset containing a different set of negatives. Training multiple SVMs on relatively small datasets is more convenient than training a single SVM on a large dataset because it can be easily paralleled. The training dataset for each SVM can be unbalanced to contain more negatives than positives. Each of the SVMs uses its own separating hyperplane to score a pair of proteins. For the following reason the score we use for prediction is the lowest one provided by any of the SVMs. Each SVM determines a linear decision boundary in the feature space induced by a kernel function. If the negatives used for training are insufficient and not representative for all negatives, the positive space defined by the decision boundary will be excessively large, causing the SVM to yield many false positives for a large test dataset. Clearly, the number of different decision boundaries increases as we increase the number of SVMs, since we train each SVM on a dataset containing the same set of positives but a different set of negatives. We want to make use of all these boundaries to expand the negative space in the feature space to reduce false positives. We do this by using the lowest score, which corresponds to choosing the decision boundary that encloses the positive space most tightly with respect to the test data point. The lowest score approach has a risk of yielding many false negatives. We will discuss this point and alternative approaches in later sections.

4.3 Datasets and measurements

4.3.1 Datasets

We used the same datasets as in Chapter 3 to examine the performances of SVMs in constructing hypothetical protein-protein interaction maps of yeasts and humans. For details on the datasets, see Subsection 3.3.1.

4.3.2 Measurements

As in Chapter 3, sensitivity (Pr), false positive rate ($FPrate$), precision (Sn), and F-measure (F) were defined by

$$Sn = \frac{TP}{TP + FN}, \tag{4.4}$$

$$FPrate = \frac{FP}{TN + FP}, \tag{4.5}$$

$$Pr = \frac{TP}{TP + FP}, \tag{4.6}$$

and

$$F = \frac{2 \times Pr \times Sn}{Pr + Sn}, \tag{4.7}$$

where TP, TN, FP, and FN represent the numbers of true positives, true negatives, false positives, and false negatives. A receiver operating characteristic (ROC) curve is a plot of the sensitivity against the false positive rate obtained when varying the threshold for prediction. We emphasize that the negative/positive ratio of the test dataset does not affect a ROC curve: random classification always yields a straight line with gradient 1, regardless of the negative/positive ratio of the test dataset.

Again, we also assessed the performance of our SVM-based approaches in a practical application by examining our classifiers abilities to predict likely interactions between protein pairs that are not known to interact. We assumed that an interaction is likely if both proteins function in the same biological processes. For this test we used a subset of GO terms that are reachable in four hops from the root node of the biological-process category. We reannotated a protein with these GO terms by tracing back the directed acyclic graph from the GO terms originally assigned to the protein. When encountering a GO term that has many parents, we traced back along all the paths. We ignored those GO terms originally assigned to a protein that are within three hops of the root node. For simplicity the semantic difference between is a and part of was ignored. After this reannotation process, a protein originally annotated with the GO term glutamate biosynthesis was, for example, reannotated with the terms cellular metabolism, biosynthesis, nitrogen compound metabolism, and primary metabolism. We then assumed that two proteins function in the same biological processes if at least one GO term matched between the two proteins. When we applied this evaluation method to the protein pairs in our training datasets, we found that 91.7% of the positives and 60.9% of the negatives in the yeast dataset function in the same biological processes, that

81.8% of the positives and 49.1% of the negatives in the human dataset function in the same biological processes, and that in each dataset the difference between the percentages of positives and negatives that function in the same biological processes was significant ($\alpha = 0.001$). The protein pairs for this test were obtained by randomly pairing the proteins 100,000 times. We then removed all pairs that were in the training dataset and all pairs that consisted of proteins that cannot be reannotated with the subset of GO terms. A total of 70,420 pairs of yeast proteins and 37,575 pairs of human proteins were used for this test.

4.4 Results

4.4.1 Cross-validation analyses on yeast protein pairs

We first used 10-fold cross-validation analyses to evaluate the performance of a single SVM in predicting protein-protein interactions in yeast. Fig. 4.1 shows ROC curves for four single-SVM-based classifiers trained on datasets containing different numbers of negatives (N/P = 1.0, N/P = 2.0, N/P = 3.0, and N/P = 4.0, where N/P denotes the negative/positive ratio). We focused on the false positive rates below 0.01 for the following reason. If there are 30,000 interactions between 6000 yeast proteins [74], there must be $6000 \times 6001/2 - 30,000 = 17,973,000$ non-interacting pairs of proteins. Thus there are 600 times as many negatives as positives, or only 0.17% of all pairs are interacting. Then even a good classifier with a false positive rate of 0.01 would yield no less than $17,973,000 \times 0.01 = 179,730$ false positives if the input data was all pairs of proteins. It is therefore important to consider very low false positive rates when assessing the performance of classifiers used for constructing an interaction map.

As shown in Fig. 4.1, the SVM trained on the dataset with the highest N/P performed best. The training time, however, increased almost linearly with the N/P ratios of the training datasets: on a computer with 600MHz CPU, the training times on the N/P = 1.0, 2.0, 3.0, and 4.0 datasets were respectively 0.4, 2.1, 3.9, and 7.6 hours. For this reason we did not use a dataset containing more than four times as many negatives as positives when we trained our SVMs.

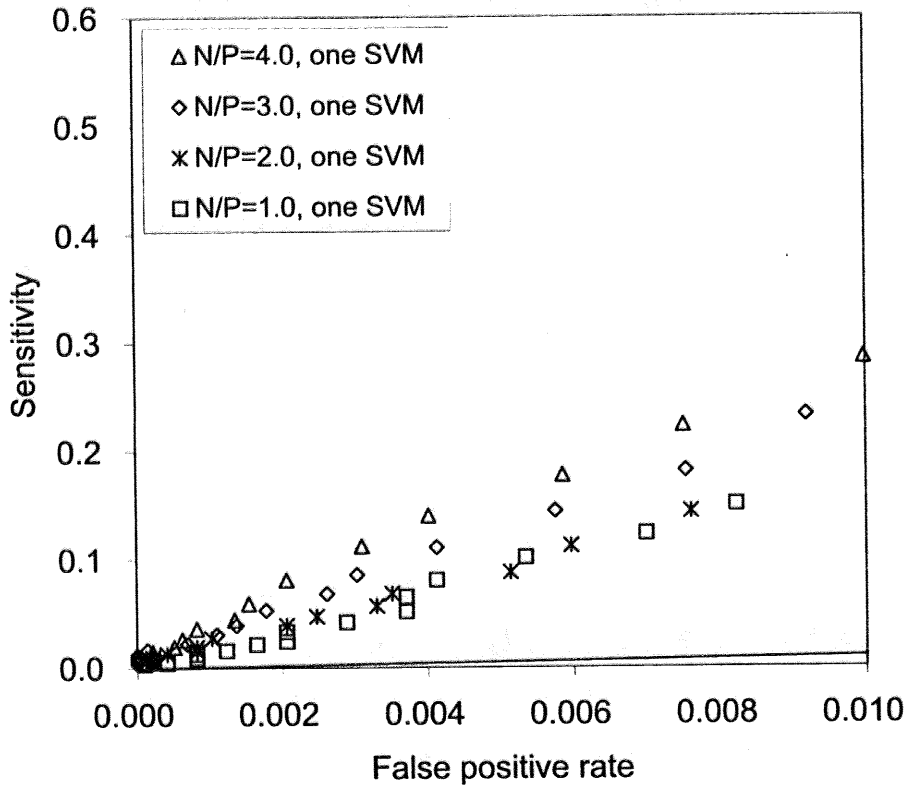

Figure 4.1. ROC curves for single-SVM-based classifiers applied to yeast-protein datasets. N/P denotes the negative/positive ratio of the dataset used for training the SVM. The black line shows the expected result of random prediction.

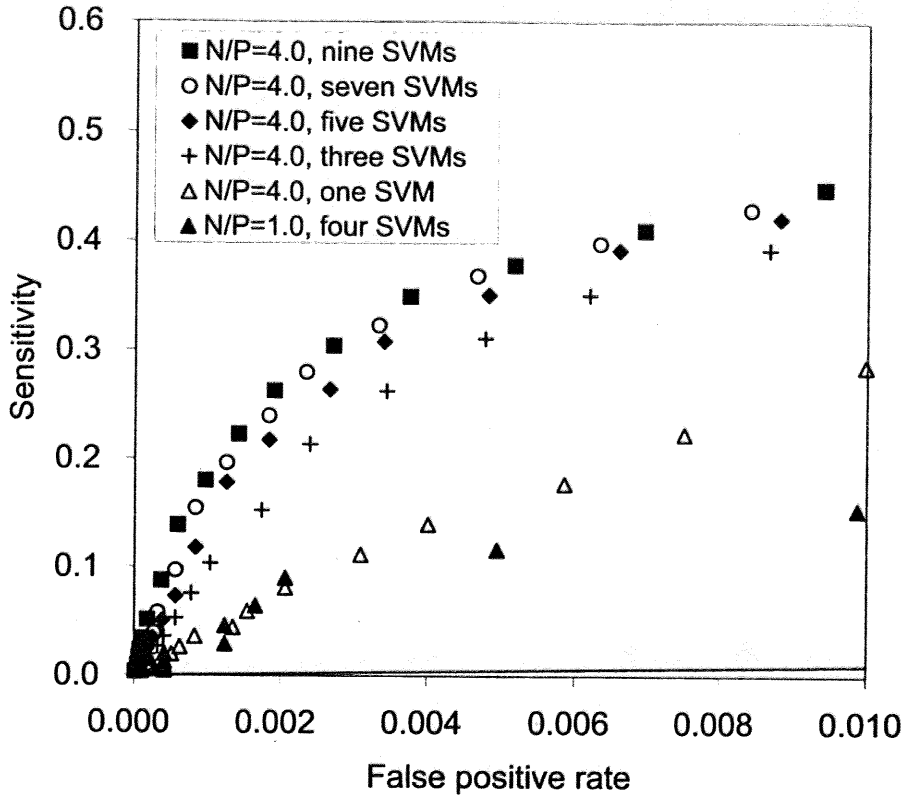

Figure 4.2. ROC curves for various lowest-score classifiers applied to yeast-protein datasets. The black line shows the expected result of random prediction.

We next examined the performance of the lowest-score approach when the number of SVMs (trained on N/P = 4.0 datasets) used for classification was varied between one and nine. As shown in Fig. 4.2, performance improved as we increased the number of SVMs. The small difference between the approach using seven SVMs and that using nine SVMs suggests that nine is the practical limit of the number of our SVMs used in the lowest-score approach.

In the above comparison, the total number of negatives used in a training procedure varied according to the number of SVMs used for classification. For example, we used $2430 \times 4 = 9720$ negatives to train one SVM and used $2430 \times 4 \times 9 = 87480$ negatives to train nine SVMs. Because training an SVM on a dataset containing more than four times as many negatives as positives requires a lot of time, we compared the performance of a single SVM trained on the N/P = 4.0 dataset with the performance obtained using the lowest score provided by four SVMs that were each trained on the N/P = 1.0 dataset. Comparing the open and filled triangles in Fig. 4.2, we see that if

the total number of negatives used in the training procedure is the same, the performance of the lowest-score approach is not necessarily better than that of the single SVM. The training times for an SVM, however, show that the lowest-score approach is far more practical than the single-SVM-based approach if more than one CPU is available. While on a computer with a 600MHz CPU it takes 7.6 hours to train a single SVM on an N/P = 4.0 dataset, on a computer with two 600MHz CPUs it would take less than an hour to train four SVMs on N/P = 1.0 datasets. It follows that the lowest-score approach using nine SVMs each trained on an N/P = 4.0 dataset would be far more practical than a single SVM trained on an N/P = 36.0 dataset.

Focusing on the SVMs trained on the N/P = 1.0 datasets, we see little difference in performance between the single SVM (Fig. 4.1, open squares) and the lowest-score approach using four SVMs (Fig. 4.2, closed triangles) even though the total number of negatives used in the training procedures was different. This may be because an improperly low score assigned by a weak SVM leads to a failure to find a true interaction, thus decreasing the sensitivity. This implies that the decrease in sensitivity can be compensated for by the large decrease in the false positive rate if the performance of each SVM used in the lowest-score approach is sufficiently high. As shown later in Figs. 4.4 and 4.5, this implication is supported by the results obtained when predicting interactions between human proteins: the performance of each SVM was relatively high and thus the lowest-score approach using four SVMs each trained on the N/P = 1.0 dataset performed far better than the single SVM trained on the N/P = 1.0 dataset and performed nearly as well as the single SVM trained on the N/P = 4.0 dataset.

The negative/positive ratio of a test dataset affects the precision defined in Eq. (4.6). Consider the following cases, for example. If test data are all true positives, even a random classifier has a precision of 1.0. The precision evaluated in most of the previous works was that of classifiers tested on datasets containing the same number of positives and negatives, in which case a random classifier would have a precision of 0.5. Practical problems are far more difficult because most of the proteins in a cell do not interact with each other. Given the negative/positive ratio N/P of a test dataset, we can estimate the precision in several situations as follows. We first define $N/P = (TN+FP)/(TP+FN)$.

From Eq. (4.4), we have $TP = Sn \times (TP + FN)$. We can thus rewrite Eq. (4.6) as

$$\begin{aligned} Pr &= \frac{TP}{TP + FP} \\ &= \frac{Sn \times (TP + FN)}{Sn \times (TP + FN) + FP} \\ &= \frac{Sn}{Sn + \frac{FP}{TP+FN}} \\ &= \frac{Sn}{Sn + \frac{TN+FP}{TP+FN} \times \frac{FP}{TN+FP}} \\ &= \frac{Sn}{Sn + N/P \times FPrate}. \end{aligned} \tag{4.8}$$

The sensitivity and false positive rate defined in Eqs. (4.4) and (4.5) are not affected by the N/P of a test dataset, and hereinafter we use Eq. (4.8) instead of Eq. (4.6) to estimate the precision obtained when testing datasets with different N/P ratios.

We first focused on the approach using nine SVMs trained on data containing four times as many negatives as positives and plotted the sensitivity-precision curves obtained when simulating the testing of datasets with various N/P ratios (Fig. 4.3). The plot indicated by circles should reflect the performance in constructing an interaction map for yeast proteins because we have estimated above that a yeast cell has 600 times as many pairs of non-interacting protein pairs as it does interacting pairs. The performance measures obtained using two different approaches at the thresholds giving the highest F-measures are listed in Table 4.1. When the test data contained the same number of positives and negatives, the performance measures differed little between the two approaches. When the test data contained 600 times as many negatives as positives, however, multiple SVMs trained on unbalanced data performed much better than a single SVM trained on balanced data.

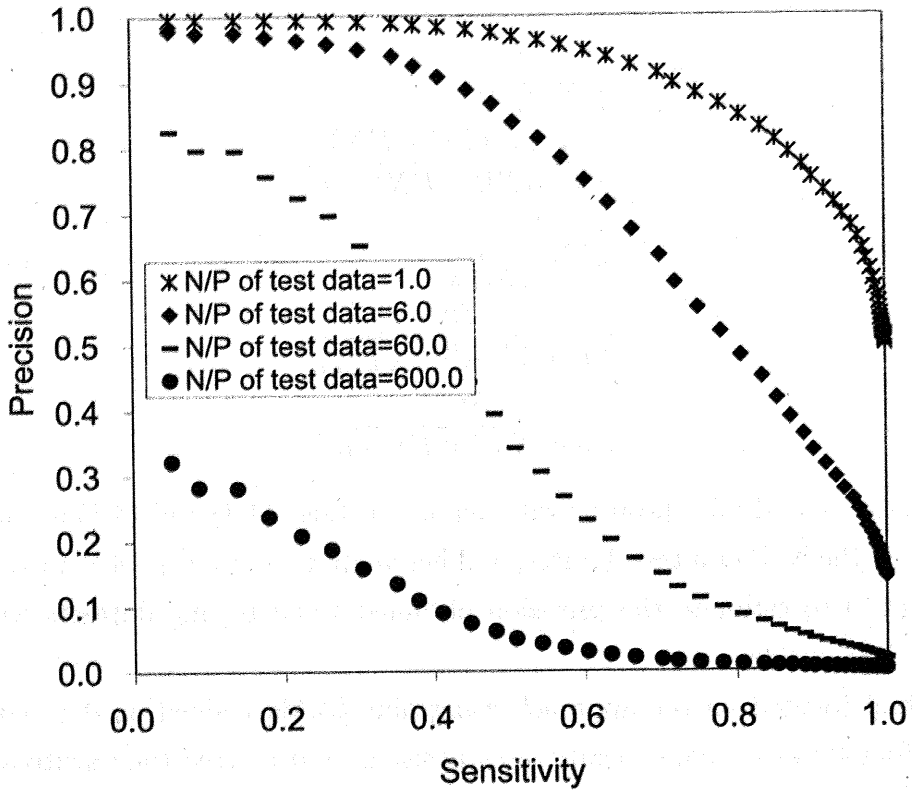

Figure 4.3. Estimated performance obtained with yeast-protein test datasets adjusted to have various negative/positive ratios. Shown are the estimated results of the lowest-score approach using nine SVMs, each trained on data containing four times as many negatives as positives. The precision was calculated using Eq. (4.8).

Table 4.1. Performance obtained at optimum thresholds with single-SVM and multiple-SVM approaches applied to yeast-protein datasets.

N/P of test data	Approach	Threshold	Precision	Sensitivity	F-measure
1.0	N/P=1.0, one SVM	0.0	0.811	0.861	0.835
1.0	N/P=4.0, nine SVMs	-1.2	0.831	0.835	0.833
600.0	N/P=1.0, one SVM	1.9	0.114	0.095	0.104
600.0	N/P=4.0, nine SVMs	0.6	0.187	0.263	0.219

The precision was calculated using Eq. (4.8), and we chose the thresholds that maximized the F-measures under the specified N/P conditions.

Table 4.2. Performance obtained at optimum thresholds with single-SVM and multiple-SVM approaches applied to human-protein datasets.

N/P of test data	Approach	Threshold	Precision	Sensitivity	F-measure
1.0	N/P=1.0, one SVM	0.0	0.832	0.859	0.845
1.0	N/P=4.0, nine SVMs	-1.1	0.859	0.852	0.855
600.0	N/P=1.0, one SVM	1.9	0.072	0.119	0.089
600.0	N/P=4.0, nine SVMs	0.7	0.291	0.300	0.296

The precision was calculated using Eq. (4.8), and we chose the thresholds that maximized the F-measures under the specified N/P conditions.

4.4.2 Cross-validation analyses on human protein pairs

The applicability of our method to the prediction of interactions between human proteins must be validated because the much larger number of human proteins makes the prediction of interactions far more difficult than it is with yeast proteins. The performance measures listed in Table 4.2 and the ROC and sensitivity-precision curves shown in Figs. 4.4, 4.5, and 4.6 are the human-protein results corresponding to the yeast-protein results listed in Table 4.1 and shown in Figs. 4.1, 4.2, and 4.3. The corresponding training times on the N/P = 1.0, 2.0, 3.0, and 4.0 datasets were respectively 16.5, 28.4, 58.5, and 64.2 hours.

Unexpectedly, the performance of SVMs trained on the human protein pairs was slightly better than that of SVMs trained on the yeast protein pairs (cf. Figs. 4.1 and 4.4). Three additional factors are important to consider. First, the training datasets for the human proteins were no more than 2.4 times larger than the training datasets for yeast proteins. Second, domains are not as diverse as proteins are: the human genome encodes at least four times as many proteins as the yeast genome does [61], but the number of domain types found in the human protein sequences is no more than 1.6 times greater than that found in the yeast protein sequences (see Subsection 4.3.1). Finally, the interaction predictions of our SVMs are based mainly on domain information. From these considerations and the prediction results we can hypothesize that over the course of evolution the number of domain combinations that mediate protein-protein interactions has not increased as much as one might have expected from the increased number of all possible pairs of proteins. The verification of this hypothesis is, however, beyond the scope of this paper.

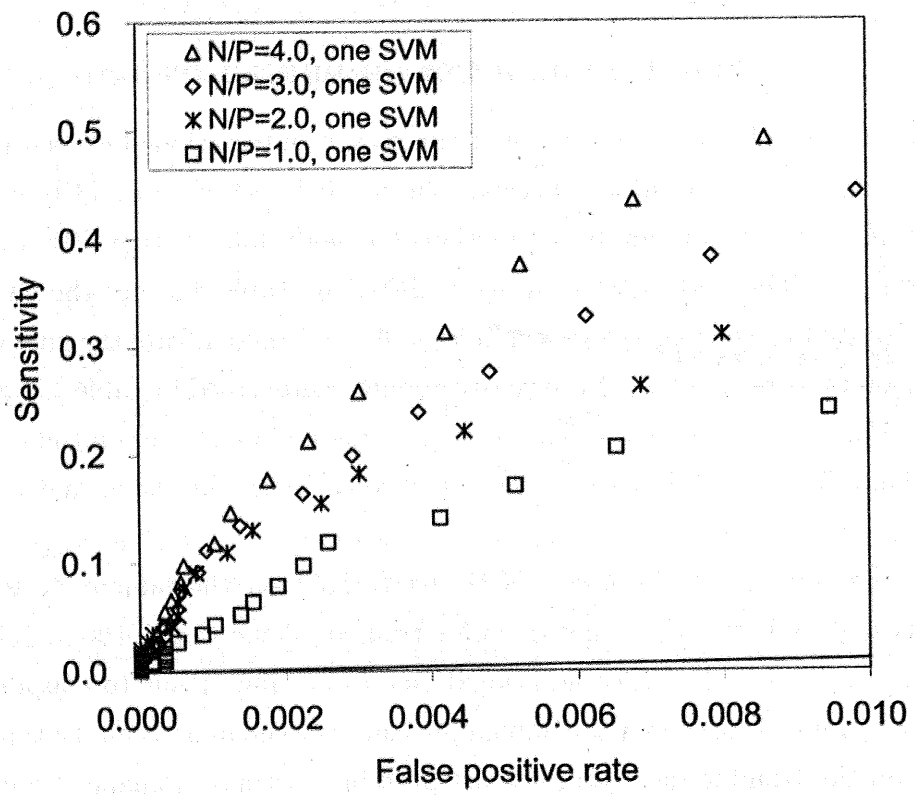

Figure 4.4. ROC curves for single-SVM-based classifiers applied to human-protein datasets. N/P denotes the negative/positive ratio of the dataset used for training the SVM. The black line shows the expected result of random prediction.

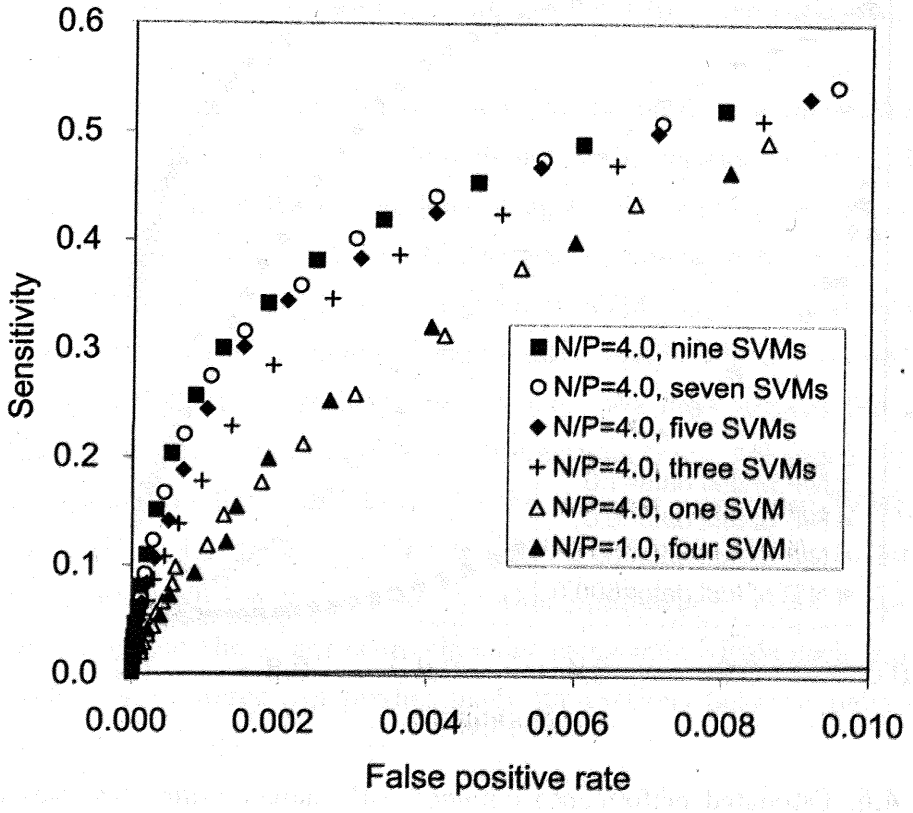

Figure 4.5. ROC curves for various lowest-score classifiers applied to human-protein datasets. The black line shows the expected result of random prediction.

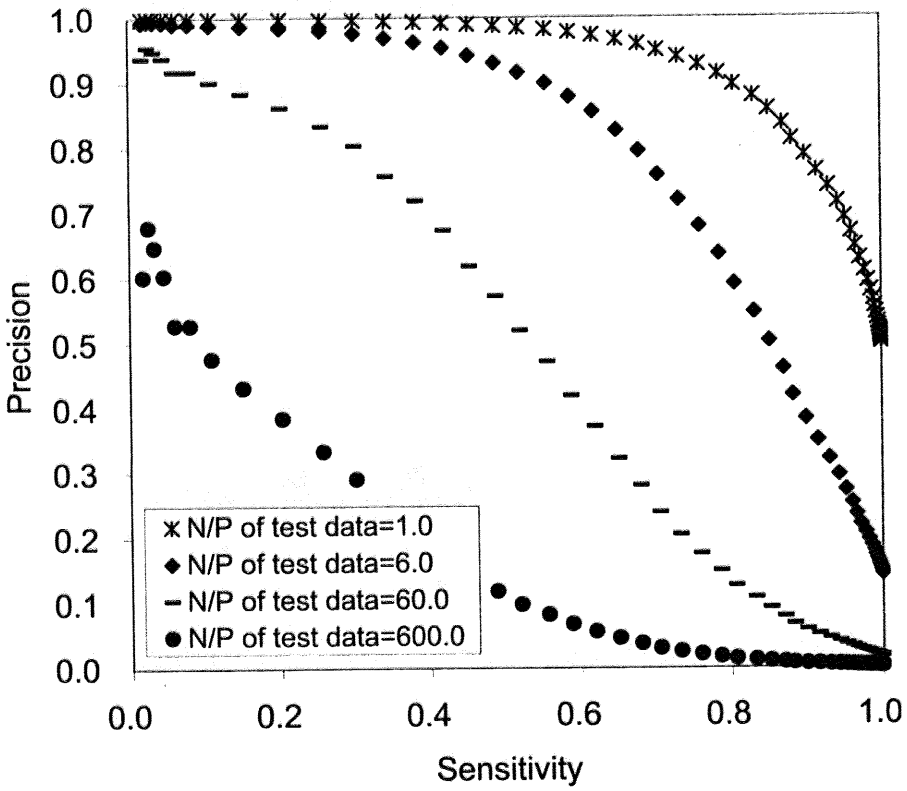

Figure 4.6. Estimated performance obtained with human-protein test datasets adjusted to have various negative/positive ratios. Shown are the estimated results of the lowest-score approach using nine SVMs, each trained on data containing four times as many negatives as positives. The precision was calculated using Eq. (4.8).

4.4.3 Predicting unknown interactions

The cross-validation analyses described in the previous sections are standard ways of estimating the actual error rate of a classifier. It is known, however, that certain negative samples lead to the overestimation of the predictive performance in cross-validation analysis [6]. In this section, we evaluate how likely the interactions predicted by our method are in terms of GO annotations. For this test we used randomly created pairs of proteins as a test dataset (see Section 4.2) that we think has the same negative/positive ratio of protein pairs that an actual cell does.

To examine whether SVM scores can be used as a measurement of the reliability of predictions, we first explored the relation between SVM scores of protein pairs and the fractions of likely interactions between those pairs. For yeast protein pairs (Fig. 4.7) as well as human protein pairs (Fig. 4.8), only the lowest-score classifier using nine SVMs, each trained on the N/P = 4.0 dataset (closed triangles), consistently assigned higher scores to pairs of proteins that are more likely to interact. We next plotted the fractions of likely interactions in the predicted interactions (Figs. 4.9 and 4.10). The pairs of proteins predicted to interact by the classifier using nine SVMs trained on the N/P = 4.0 dataset (closed triangles) were more likely to interact than the pairs predicted to interact by other approaches. From these results we conclude that the lowest-score classifier using nine SVMs each trained on the N/P = 4.0 dataset can predict likely interactions between pairs of proteins created randomly.

4.4.4 Predicting likely interactions among high-throughput interactions

High-throughput experimental systems for detecting binary protein-protein interactions, such as the yeast two-hybrid system, are known to yield many false positives. Predicting likely interactions among high-throughput interactions is therefore another important problem that needs to be solved if we are to obtain reliable interaction maps. Examining the applicability of our method to this problem, we used four independent test datasets: two sets of interactions between yeast proteins that were reported by Uetz *et al.* [70] and Ito *et al.*

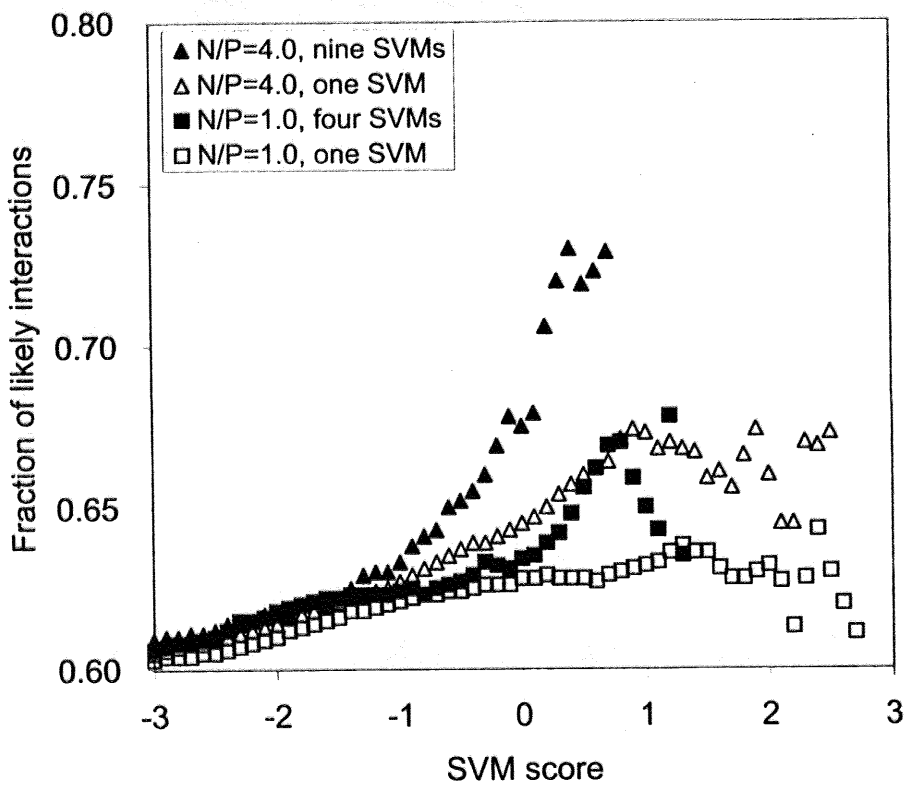

Figure 4.7. Fraction of likely interactions between pairs of yeast proteins versus the SVM scores of those pairs.

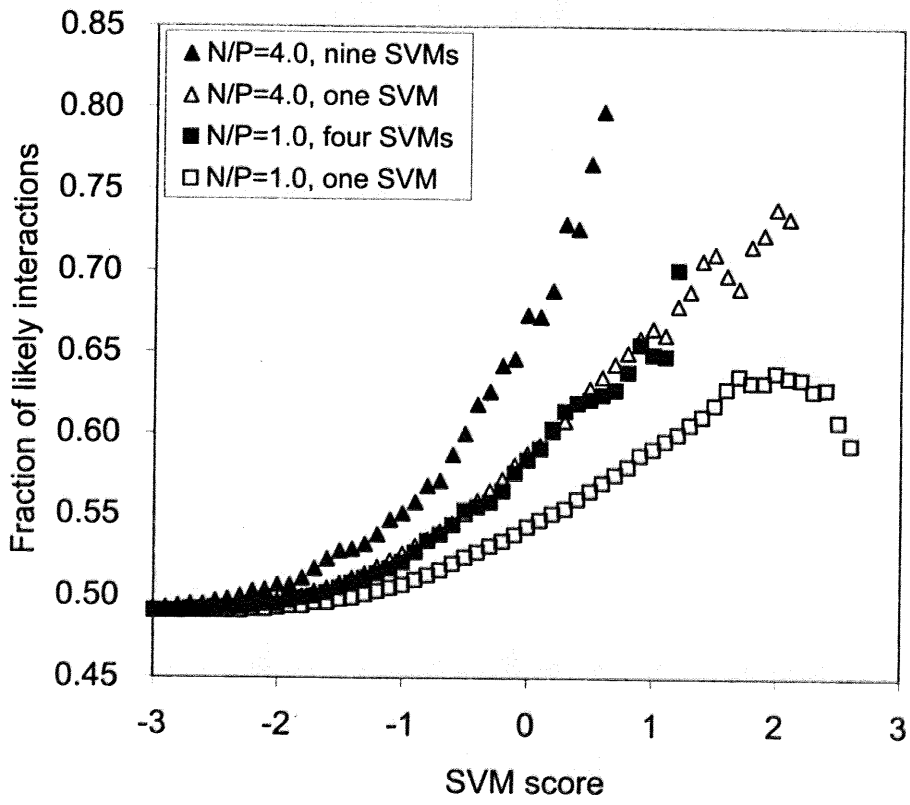

Figure 4.8. Fraction of likely interactions between pairs of human proteins versus the SVM scores of those pairs.

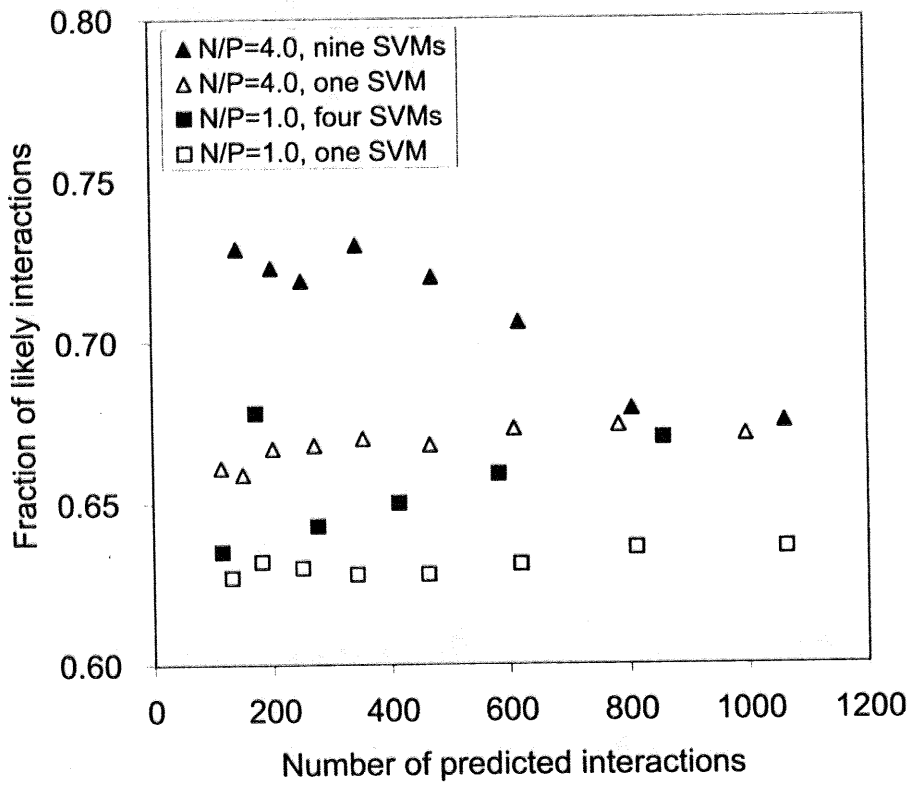

Figure 4.9. A comparison of the predictive abilities of different approaches applied to yeast protein pairs. Because 70,420 pairs of yeast proteins are estimated to contain roughly $70,420 \times 0.0017 = 120$ interacting pairs, here we show the results for only the 1200 highest-scoring pairs.

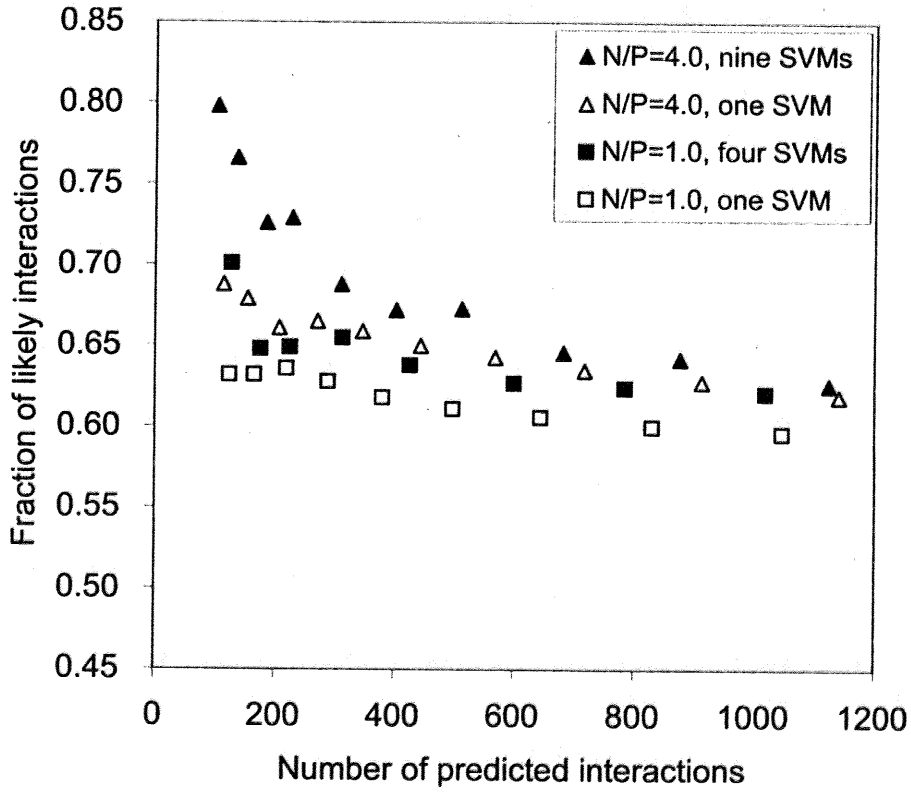

Figure 4.10. A comparison of the predictive abilities of different approaches applied to human protein pairs. Note that although here, as in Fig. 4.9, we show the results for the 1200 highest-scoring pairs of proteins, far less than 1200 of the 37,575 random pairs of proteins used in this test are expected to actually interact.

Table 4.3. Performance of our approach evaluated on high-throughput interactions between yeast proteins.

Test data	Condition	Number of data	Number of predicted interactions	Fraction
Uetz *et al.*	High-throughput	167 (269)	46 (138)	0.275 (0.513)
	Protein Array	103 (151)	55 (103)	0.534 (0.682)
Ito *et al.*	Non-core	1240 (1295)	340 (428)	0.274 (0.331)
	Core	255 (377)	112 (227)	0.439 (0.602)

The performance of the lowest-core classifier using nine SVMs, each trained on unbalanced data, was evaluated on the high-throughput interactions reported by Uetz *et al.* [70] and Ito *et al.* [35]. Assuming that input datasets contained roughly the same number of positives and negatives, we set the threshold at a SVM score of -1.2 according to the results listed in Table 4.1. The protein-protein interactions used for training SVMs were removed from test datasets. The results on test datasets including these interactions are shown in parentheses for reference. The numbers of interactions used for this test were fewer than the number of interactions originally reported because our method requires a protein to have at least one known domain. Note that the two datasets reported by Uetz *et al.*, "High-throughput" and "Protein Array," are not mutually exclusive. After removing the interactions used for training SVMs, however, we found only one interaction that was detected by both approaches.

[35] and two sets of interactions between human proteins that were reported by Rual *et al.* [62] and the Genome Network Project (GNP) administered by the Ministry of Education, Culture, Sports, Science and Technology of Japan (http://genomenetwork.nig.ac.jp/index_e.html). For this test we used a lowest-score classifier using nine SVMs, each trained on an N/P = 4.0 dataset and calculated the fraction of predicted interactions in each high-throughput dataset (Tables. 4.3 and 4.4).

Uetz *et al.* developed two different yeast two-hybrid systems, a high-throughput screen based on an activation-domain library (High-throughput) and a protein array screen (Protein Array). A comparison with a compilation of literature-cited interactions indicated that the interactions identified by the latter approach were more reliable (Table. 2 in ref. [70]). Our prediction results are consistent with this: the fraction of predicted interactions was higher in the latter dataset (Table 4.3). Ito *et al.* designated their core dataset on the basis of IST hits, which was considered to be more reliable than the non-core dataset (Core and Non-core) [35]. We confirmed that the core data was more

Table 4.4. Performance of our approach evaluated on high-throughput interactions between human proteins.

Test data	Condition	Number of data	Number of predicted interactions	Fraction
Rual *et al.*	Y2H	1593 (1614)	655 (676)	0.411 (0.419)
	LCI	3043 (3612)	2400 (2945)	0.789 (0.815)
GNP	Y2H	167 (168)	106 (107)	0.635 (0.637)

The performance of the lowest-score classifier using nine SVMs, each trained on unbalanced data, was evaluated on the high-throughput interactions reported by Rual *et al.* [62] and the Genome Network Project (http://genomenetwork.nig.ac.jp/index_e.html). Assuming that input datasets contained roughly the same number of positives and negatives, we set the threshold at a SVM score of -1.1 according to the results listed in Table 4.2. The protein-protein interactions used for training SVMs were removed from test datasets, and the results on test datasets including these interactions are shown in parentheses. For the data reported by Rual *et al.*, we found 61 interactions that appeared in both the Y2H and LCI datasets after removing the interactions used for training SVMs, 49 of which were predicted as positive by our approach (fraction: 0.803).

likely to be predicted by our approach (Table 4.3). A high-throughput experiment to detect protein-protein interactions in human has been performed by Rual *et al.* The initial dataset contained two types of interactions: those identified by their yeast two-hybrid screens (Y2H) and those extracted from literature (LCI) [62]. We observed higher fraction of predicted interactions in LCI than in Y2H (Table 4.4). This difference may be partially due to the "inspection bias" in our training datasets. Meanwhile, 80.3% of the interactions present in both Y2H and LCI datasets but not in our training dataset were predicted by our method. Furthermore, our method predicted 63.5% of yeast two-hybrid interactions identified by the GNP (Table 4.4). This implies that Y2H dataset reported by Rual *et al.* indeed contained many false positives. These results indicate that our approach can contribute to obtaining reliable interaction maps by eliminating erroneous interactions from high-throughput datasets, as well as by predicting interaction de novo. We list all predicted interactions in these datasets in Tables A.1 to A.4.

4.4.5 Examples of predicted human protein-protein interactions

We have been predicting human protein-protein interactions using nine SVMs trained on N/P=4.0 dataset. The predicted interactions can be used for several purposes such as predicting protein functions, functional modules, disease-related proteins, and relationship between diseases. In this subsection, we show an example that a plausible hypothesis on the mechanism of Alzheimer's disease (AD) can be proposed by mapping the predicted and known protein-protein interactions onto a known causal pathway of AD.

We obtained the known causal pathway of AD from KEGG [38]. The pathway was shown in Fig. 4.11. AD is a common age-related brain disorder that progressively destroys a person's memory and intellect. The critical characteristic of the brain in AD is the presence of amyloid plaques in the spaces between the nerve cells. The protein basis of these amyloid plaques, Aβ 40/42, is formed by cleaving amyloid precursor protein (APP) by β-secretase (BACE1) and γ-secretase, and this processing is believed to be regulated by presenilins (PSEN1 and PSEN2) [7, 33, 52, 68, 77]. On the other hand, non-Aβ component of Alzheimer's disease amyloid (NAC) is another important component of amyloid plaques. NAC is an amino acid fragment generated from its precursor protein, α-synuclein (SNCA), whose defects are also known to be involved in the pathogenesis of Parkinson disease (PD) [65]. However, the regulation mechanism of the processing has not yet been elucidated completely.

To make a hypothesis on which proteins regulate the processing of SNCA, we mapped onto the pathway the protein-protein interactions obtained from three different sources: currently predicted interactions by our method, interactions derived from HPRD [59], and interactions reported by Rual *et al* [62]. These interactions were shown in different colors in Fig. 4.11 according to the data sources.

We first observed that APP protein was predicted to interact directly with PSEN1 protein, and this interaction was also found in the dataset reported by Rual *et al* [62]. On the other hand, SNCA protein was predicted to interact with its two close homologues, β- and γ-synuclein (SNCB and SNCG). In particular, SNCB is known to be abundantly expressed in neurofibrils in the

brain in AD [17], and is believed to act as a negative regulator of α-synuclein [67]. We further found that SNCB protein is predicted to interact with PSEN2 protein which is postulated to regulate APP processing as described above. From these observations, we can hypothesize that presenilins are associated with synuclein family, and that they regulate the processing of SNCA protein to NAC as well as that of APP protein to Aβ 40/42.

Experimental verification of the hypothesis is beyond the scope of this thesis. Nevertheless, hypothesis such as that described above will help researchers gain a first insight into their experimental targets, and will accelerate our understanding of biological systems and disease mechanisms. Although here we focused on AD, hypotheses on the mechanisms of other diseases, such as PD, will be generated analogously. Therefore, in the future, not only the mechanism of each disease but also the relationships between diseases will be explored by using our predicted protein-protein interactions.

4.5 Disucussion

Feature selection is an important step for prediction. Previous studies have explored the use of several protein features such as charge, hydrophobicity, surface tension [10], domains [14, 22, 5] and k-mers of amino acids [5, 48], and even non-sequence-based features such as GO annotations [5]. Ben-Hur *et al.* reported that their method involving the GO kernel correctly predicts 80% of the actual interactions and has a false positive rate of 1% [5]. We did not use the information on GO annotation because many proteins, especially mammalian proteins, have no GO annotations and some GO annotations have been assigned to proteins on the basis of the physical interaction partners of those proteins (GO annotations with evidence code IPI). Regardless of the protein features used, however, a training dataset with as many non-interacting pairs as there are interacting pairs does not adequately represent the actual number of non-interacting pairs. Taking our SVM-based method as an example, we have demonstrated that the predictive performance can be improved by simply increasing the number of non-interacting pairs in the training dataset. This, however, increases the training time significantly and thereby makes it hard to tune the parameters of SVMs and to update the training dataset when new

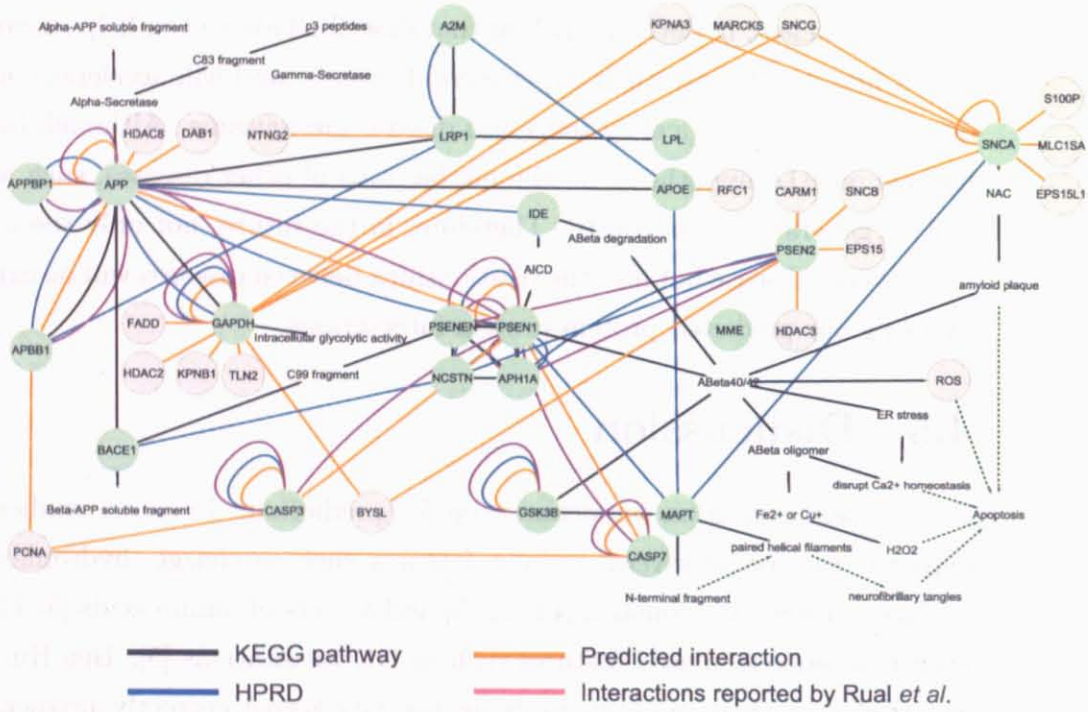

Figure 4.11. The causal pathway of Alzheimer's disease was obtained from KEGG [38]. The green nodes represent the proteins that appear in the original KEGG pathway diagram, and the gray nodes represent the proteins that do not appear in the original pathway but are predicted to interact with the proteins shown in green. The edges represent interactions or other relations, and are differently colored based on the data sources. The edge line types are the same as defined in KEGG.

interactions are identified. In this work we have shown that the lowest-score approach using multiple SVMs, each trained on an unbalanced dataset, can improve the predictive performance in practical situations. As shown in Fig. 4.2, the lowest-score approach may not be better than the single SVM when the total number of negatives used for the training is the same. If more than one CPU is available, however, and if the SVM performance is sufficiently high, the lowest-score approach is useful to improve the prediction of protein-protein interactions by enabling the use of a large number of non-interacting pairs of protein (Tables 4.1 and 4.2).

Several alternative approaches can be considered. One is to assemble an unbalanced dataset that contains more than four times as many negatives as positives. Our preliminary tests revealed, however, that a significant amount of time was then required for the training procedure. In the lowest-score approach, SVM scores can be weighted according to the predictive ability of each SVM. This seemed to make little difference for our SVMs because their performances differed only slightly in a 10-fold cross-validation test. Other techniques such as boosting, transductive inference methods such as Transductive Support Vector Machines [37], and automatic negative data collection schemes such as PEBL [82] can be used to devise a better classifier. We found, however, that none of these methods trained classifiers in a practical length of time: less than a month. The lowest-score approach using multiple SVMs is simple but practical for predicting the interactions between pairs of proteins in large datasets.

Constructing protein-protein interaction maps for higher organisms, especially mammals, is of great significance because relatively little information about mammalian protein-protein interaction is available from public databases. Currently, we have been constructing a hypothetical protein-protein interaction maps of humans. We have also set up a Web-based service called the Protein-protein Interaction Prediction Server (PIPS) that can predict physical protein-protein interactions between yeast, mouse, and human proteins. In the next chapter, we will discuss the details of this Web service.

Chapter 5

Web application

5.1 Introduction

Information on physical protein-protein interactions provides us with many clues for analyzing protein functions in cellular processes. For a detailed and comprehensive understanding of a protein of interest, its interaction partners should be identified from a large number of proteins in the organism. The key to reducing time, cost and human resources is for us to predict likely interactions and gain initial insights into nature prior to conducting biological experiments. To facilitate these pre-experimental tasks, we developed a Web server called the Protein-protein Interaction Prediction Server (PIPS), which can predict the physical interactions between two proteins.

Supervised machine learning on protein-protein interaction data has been recognized as a prospective approach to accurate predictions [5, 14, 22, 48]. Two issues, however, are worth considering to use the approach in practice. The first is that most previous methods utilizing supervised machine learning have been developed to predict interactions in yeast. PIPS is a novel server that provides a way of applying Support Vector Machines (SVMs), one of the most intensively studied supervised machine learning methods, to predicting interactions in mice and humans as well as in yeasts. The second is the difficulty of sampling good negative examples to train classifiers. Based on the results described in the previous chapter, PIPS predicts likely interactions with nine SVMs trained on datasets containing four times as many non-interacting pairs of proteins as interacting pairs of proteins. The services are available from http://prime.ontology.ims.u-tokyo.ac.jp:8081/cgi-bin/PIPS.cgi.

5.2 Server description

5.2.1 Server Architecture

Fig. 5.1 shows the top-level Web page of PIPS. Based on the existence/absence of domains and amino acid compositions, PIPS can predict protein-protein interactions in yeasts, mice, and humans. For each organism, the system has implemented nine SVMs, each trained on dataset containing four times as many non-interacting pairs of proteins (negatives) as interacting pairs of proteins (positives). Given two FASTA-formatted protein sequences, the system first searches domains retained in the query proteins by running hmmpfam [30]. It subsequently assembles a feature vector, classifies the vector with SVMs, and finally creates a Web page with prediction results.

PIPS can perform homology-based prediction, optionally, in conjunction with PRIME [42]. PRIME stores interactions automatically extracted from Medline abstracts through natural language processing techniques as well as those imported from the major public databases such as DIP, BIND, and MIPS. For a query protein pair, the homology-based method implemented by the system predicts interactions by searching all the possible pairs of their homologs in PRIME. All pairs of homologs that are known to interact and registered in PRIME are displayed in the result Web page along with a result of SVM-based prediction. For details of the homology-based prediction, see Subsection 3.2.4.

5.2.2 Usage

PIPS is quite simple to use. The steps required for predicting an interaction between two proteins are to select an organism and to copy and paste two FASTA-formatted amino acid sequences. The system starts when the user pushes the "Predict!" button. The Web browser must be kept open until the result page is displayed. Alternatively, by inputting an E-mail address into the corresponding box on the top page, he or she can choose to receive an E-mail that informs him or her about the URL for the result page when calculation is complete. When the "Perform homology-based prediction" box is checked, PIPS performs homology-based prediction simultaneously with the SVM-based prediction. He or she can select the interaction type (direct and/or indirect)

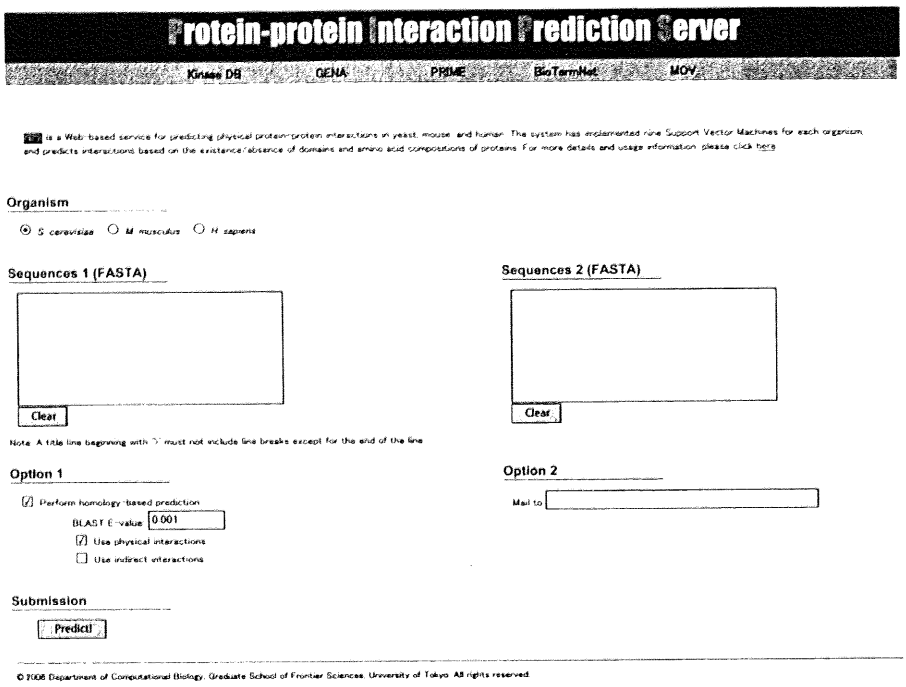

Figure 5.1. Top page of PIPS.

to be displayed and can change the BLASTP E-value for searching homologs. This option is especially useful for predicting interactions between proteins containing no domains and for searching for evidences from Medline abstracts or from public databases. This function is similar to that of InterWeaver [83]. A sample of a result Web page is shown in Fig. 5.2

5.3 Discussion

Most of the previous studies using methods based on supervised machine learning to predict protein-protein interactions have used training datasets containing approximately the same numbers of positives and negatives. It has previously been suggested, however, that with these training datasets even good classifiers tend to yield many false positives when they are applied to an input dataset containing far more non-interacting pairs of proteins than interacting pairs of proteins [48]. As discussed in Chapter 4, this is partially because negatives that is sampled as much as positives for training classifiers cannot represent all the negatives that should be considered. PIPS tackled this problem by compensating for the information on negative examples, i.e., by

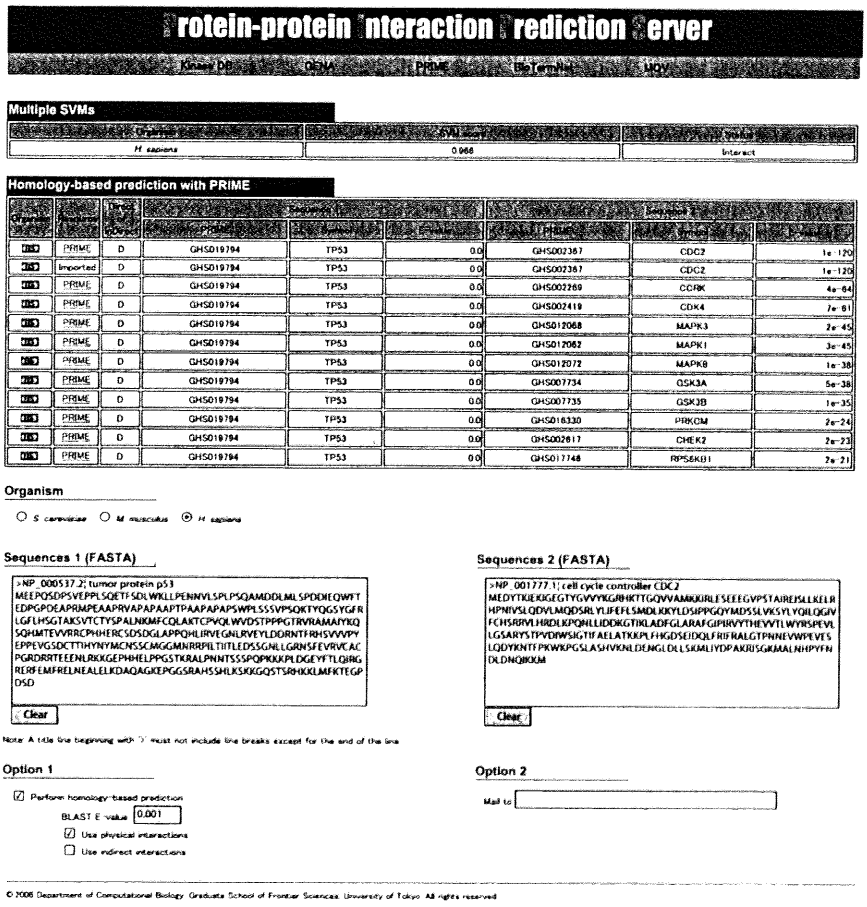

Figure 5.2. A sample of a result Web page. The query pair is an amino-acid sequence of tumor protein TP53 and of cell cycle controller CDC2 of humans. We checked the "Perform homology-based prediction" box so that the prediction results of the homology-based method was shown on the Web page.

implementing nine SVMs each trained on data containing four times as many negatives as positives. As we can see from the results shown in Figs. 4.2 and 4.5, this approach improves the ability of predicting interactions.

The server PIPS is novel in that it is able to predict physical interactions between mammalian proteins as well as that between yeast proteins. Currently, few servers are publicly available to predict the interactions in mice and humans. An exception is OPHID developed by Brown *et al.* [11]. An advantage of their the system is that it provides several evidences such as domain-domain co-occurrence, gene co-expression, and GO annotations with the prediction results. The system predicts interactions based on the homology-based method, however, which shows poorer performance than our SVM-based method does (Figs. 3.4, 3.5, 3.7, and 3.8).

Some defects of the PIPS are worth noting. First, PIPS cannot predict interactions between proteins that do not have known Pfam domains. This restriction reduces the numbers of target proteins to 58.9%, 66.7%, 65.7%, respectively for yeasts, mice, and humans. However, PIPS can eliminate this drawback by performing the homology-based prediction, which does not require a protein to have domains. Second, PIPS requires few minutes to predict interactions especially due to the domain search procedure. PIPS provides an E-mail option as a solution for this drawback. Another possible solution is to provide all predicted protein-protein interactions as a database. Such a database will significantly reduce search time and will be useful for users who wish to know the interactions between characterized proteins. To this end, we have now been predicting interactions between human proteins as an opening activity. Nonetheless, we emphasize that it is significant to run PIPS using FASTA-formatted protein sequences as a query because it provides a way to predict interactions between uncharacterized, newly discovered, or even artificial proteins. Third and finally, PIPS can predict protein-protein interactions only in yeasts, mice, and humans. Predicting protein-protein interactions in other model organisms such as chimpanzee, rat, and guinea pig will have significant impact for pharmaceutical industry, because few interaction data are currently available from public databases. Since cross-species prediction is applicable for our SVMs when the evolutionary distance is short between the source and target organisms (Table 3.7), prediction of interactions in these

mammals will be possible by using the SVMs trained on human protein pairs. We will provide the predicted protein-protein interactions in these mammals if sufficient numbers of protein-protein interactions are made available for testing the performance of our SVMs.

Chapter 6

Conclusions and future work

We have developed a new SVM-based method for predicting physical protein-protein interactions in yeasts, mice, and humans. The details were presented in Chapter 3. The method is novel in that it takes into account the protein-protein interactions mediated by more than two domains, and that several protein features such as amino acid compositions and subcellular localizations can easily be combined into feature vectors. A cross-validation analysis on a yeast-protein dataset showed that the highest F-measure of 0.788 was obtained by combining the features "domain," "amino acid composition," and "subcellular localization," which was more accurate than that of the methods reported so far. The method can also be used to assess the reliability of the interactions detected by error-prone systems. Although in this study we focused mainly on the assessment of protein-protein interactions detected by yeast two-hybrid systems, our method may be used to validate other high-throughput interaction data such as those automatically extracted from literature. The present method is also applicable to the prediction of protein-protein interactions in mammals.

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 performances of the methods proposed so far, including the one described in Chapter 3, are not good enough for this task, because even a good classifier yields a huge number of false positives if input data is all the pairs of proteins in a given organism. We have therefore developed a method based on multiple SVMs that uses more negatives than positives. The details were described in Chapter 4. Using our SVMs developed in Chapter 3, we

first showed that the negatives used in the previous studies cannot adequately represent all the negatives that need to be taken into account, thus causing the classifiers to yield many false positives. We then showed that, if more than one CPU was 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. We further demonstrated that our multiple-SVM-based method can also be used to extract likely interactions from high-throughput interactions, which is another important issue in obtaining a part of reliable interaction maps from existing data. Currently, we have been constructing a hypothetical protein-protein interaction map of humans. A resulting map will provide us with many clues for the understanding of biological systems and disease mechanisms.

A Web-based service described in Chapter 5 provides a way of applying the multiple-SVM-based method to the prediction of interactions between proteins of interest. The server can facilitate experimenters to detect true protein-protein interactions efficiently. The main drawback of the SVM-based method is that it requires the information on domains. As for the Web service, a homology-based method implemented by the system is especially useful for predicting interactions between proteins containing no domains.

Our future work encompasses three directions. The first direction is to improve the performance in predicting a comprehensive set of protein-protein interactions in an organism. The task is quite difficult as we first estimated in Chapter 4, and many more efforts are necessary to further improve the predictive performance. The performance is expected to improve if more information on protein-protein interactions and subcellular localizations of proteins is made available for public, as indicated in Chapter 3. Another possible way to improve the performance is to make use of the feedback from experimenters with the scheme of active learning. In this approach, test protein pairs are first scored using our SVMs, and those whose data points fall into the margin area (the pairs that have the scores above -1 and below +1) of each of the SVM and those that are predicted inconsistently between SVMs are experimentally examined. Then the training datasets are updated, and the SVMs are retrained to score new test protein pairs. The above steps are repeated until acceptable performance is achieved. This procedure should efficiently improve

the performance of SVMs because we can selectively obtain the new data for training that are difficult to classify by the SVMs trained on the current data. For this approach to work, it is inevitable to collaborate with experimenters. Improvement in performance makes it possible to apply our method to more specific problems. For example, roles of splice variants in health and disease will be explored by predicting and analyzing the difference in the interaction partners between splice variants. It will be also possible to investigate the difference in function in biological processes between very similar proteins by predicting and comparing the interaction partners of the two proteins.

The second direction is to predict interactions between proteins from different organisms. This task is inevitable to analyze the mechanisms underlying pathogenesis. For example, predicting interactions between virus proteins and human proteins will explore how the virus proteins affect the biological systems of humans. The method presented in this thesis will be able to apply to this task if sufficient number of interactions between proteins from different organisms is made available for training SVMs.

The third and final direction of our future work is to predict the causes, or rules, of interactions. The method presented in this thesis is mainly devised so that biologists can gain insight into and frame hypotheses on biological systems of interests from a set of protein-protein interactions and, as such, it is not suitable for inferring the cause of each interaction. An approach to apply our method to this task is to predict interactions between sequence fragments so that we can infer the key peptides that are responsible for the interactions between full-length proteins. For this aim, the SVMs must be trained on a large set of interactions between sequence fragments, which are not publicly available so far.

To explore relationships between biomolecules, for example, protein-protein interactions, is indispensable for a systematic understanding of biological systems. Challenges for a comprehensive prediction of protein-protein interactions are just beginning. The predicted interactions and a resulting map will accelerate and deepen our understating of biological systems and mechanisms of diseases and disorders.