Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 858256, 7 pages
http://dx.doi.org/10.1155/2015/858256

Research Article

Prediction of Protein-Protein Interaction By Metasample-Based Sparse Representation

Xiuquan Du,[1,2] Xinrui Li,[2] Hanqian Zhang,[2] and Yanping Zhang[1,2]

[1] *Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei, Anhui 230601, China*
[2] *School of Computer Science and Technology, Anhui University, 111 Jiulong Road, Hefei, Anhui 230601, China*

Correspondence should be addressed to Xiuquan Du; dxqllp@163.com and Yanping Zhang; zhangyp2@gmail.com

Received 23 March 2015; Accepted 28 May 2015

Academic Editor: Zenghui Wang

Protein-protein interactions (PPIs) play key roles in many cellular processes such as transcription regulation, cell metabolism, and endocrine function. Understanding these interactions takes a great promotion to the pathogenesis and treatment of various diseases. A large amount of data has been generated by experimental techniques; however, most of these data are usually incomplete or noisy, and the current biological experimental techniques are always very time-consuming and expensive. In this paper, we proposed a novel method (metasample-based sparse representation classification, MSRC) for PPIs prediction. A group of metasamples are extracted from the original training samples and then use the l_1-regularized least square method to express a new testing sample as the linear combination of these metasamples. PPIs prediction is achieved by using a discrimination function defined in the representation coefficients. The MSRC is applied to PPIs dataset; it achieves 84.9% sensitivity, and 94.55% specificity, which is slightly lower than support vector machine (SVM) and much higher than naive Bayes (NB), neural networks (NN), and k-nearest neighbor (KNN). The result shows that the MSRC is efficient for PPIs prediction.

1. Introduction

Protein-protein interactions are a hot research topic of bioinformatics. Proteins form protein-protein complexes and perform different biological processes by the interaction between protein and protein. PPIs play important roles in most cellular processes including regulation of transcription and translation, signal transduction, and recognition of foreign molecules [1]. So far, many experimental methods have been explored for detecting PPIs, including two-hybrid systems, which detect both transient and stable interactions [2, 3], mass spectrometry, which is used to identify components of protein complexes [4], and protein chip technology [5], which solidifies some proteins already known to us on a chip, and then uses the chip to predict the interactions of proteins; the advantages of these methods are easy to manipulate, and the results generated from these experimental methods are intuitive and authentic; however, such experiments for high throughput data are impossible.

Currently, a number of computational methods have been widely exploited for the prediction of PPIs. These computational methods [6] can be roughly divided into sequence-based [7–9], structure-based [10–12], and function annotation-based [13–15] methods. The advantage of sequence-based methods is not requiring expensive and time-consuming processes to determine protein structures. Martin et al. [16] used a novel description of interacting protein by extending the signature descriptor to predict PPIs. Bock and Gough [17, 18] attempt to solve the classification problem based on SVM with several structural and physiochemical descriptors. The pseudoamino acid composition approach [19, 20] was used to predict PPIs in a hybridization space by Chou and Cai [21]. The autocorrelation descriptor with SVM was used to predict PPIs by Guo et al. [22] and when performed on the PPI data of yeast *S. cerevisiae*, it achieved a very promising prediction result. Zhang et al. [23] used pairwise kernel support vector machine to predict PPIs. There are already many ways to predict PPIs, but these methods are

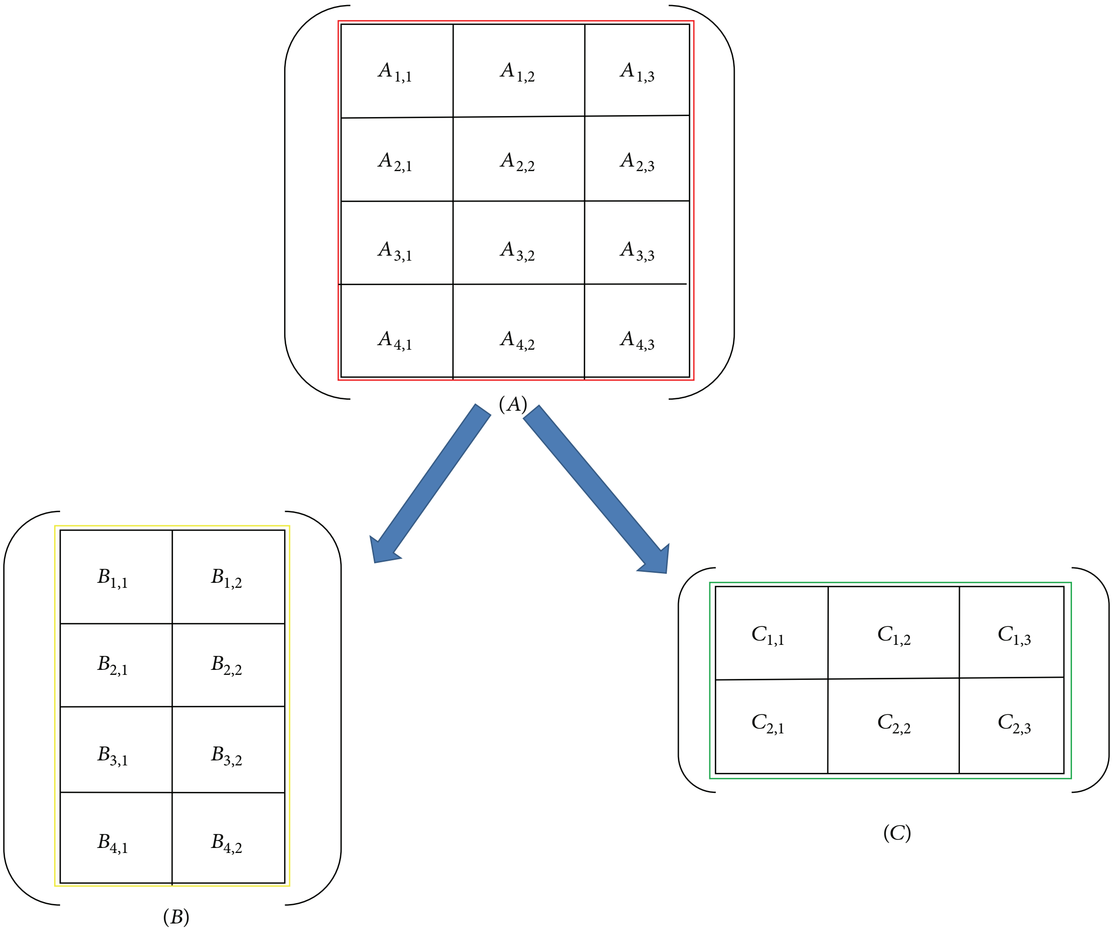

Figure 1: The metasample model of protein-protein interactions.

not efficient and reliable to a certain extent. Moreover, most of them have not adequately taken the local environment of residues into account.

Sparse presentation which is inspired by the recent progress of l_1-norm minimization based methods is a powerful data processing method and the l_1-norm minimization based methods include basis pursuing [24], compressive sensing for sparse signal reconstruction [25–27], and least absolute shrinkage and selection operator (LASSO) algorithm for feature selection [28]. The SR method presents a test sample in terms of the training samples of the same category. To discover the SR coefficient vector, l_1-regularized least square [29] should be used. A training procedure is used to create a classification model for testing in the common learning methods. Different from that, the sparse representation approach does not include separate training and testing stages. The SR methods present the PPIs test dataset as a sparse linear combination of the original training samples, and the representation error over each class is regarded as an indicator. Nevertheless, due to the fact that the original PPIs training samples do not contain the intrinsic structural information of the data, the metasample [30, 31] must be more effective for PPIs prediction than the original training samples.

The metasample can grasp the intrinsic structural information of the data, which present protein-protein interactions as a linear combination. The metasample can be obtained by using singular value decomposition (SVD) from the original PPI data. The l_1-regularized least square is used to find the SR coefficient vector, and classification is achieved on the metasamples by using a discriminating function of the SR coefficient vector.

Here, we use the sparse representation classification (SRC) [32] method with metasample for PPIs prediction; the approach is named as metasample-based sparse representation classification (MSRC) [33].

2. Methods

2.1. Metasample of PPIs Data. Normally, metasamples which can receive the inherent information are extracted from the original sample and defined as a linear combination of several samples. Through the matrix decomposition, Figure 1 illustrates the original matrix is converted into the following two matrices:

$$A \sim BC. \qquad (1)$$

The PPIs data are represented as matrix A by preprocessing. Each row represents sample and each column represents feature. The original matrix can be converted into two matrices, where B is of size $m \times q$ and C is of size $q \times n$; each of the q columns defines a metasample. Thus a lot of information which may express the implicit characteristic of data is obtained.

For metasamples, it can be extracted based on SVD which is used for matrix decomposition, and it is expected to acquire some implicit information of PPIs data for classification.

SVD is one of the important matrix decompositions in linear algebra. SVD converts the original matrix into a feature matrix and a diagonal matrix which consisted of feature value. The feature value from smallest to largest is arranged in sequence in the diagonal matrix. The researchers use several columns of data to arrange front in the feature matrix. In other words, for a matrix with high dimension, SVD performs a linear transformation on the matrix.

2.2. Sparse Representation of Test PPI Samples. In fact, PPIs prediction is a binary classification problem. Normally, training dataset of PPIs is represented by $m \times n$ matrix A with each sample being a row and each feature being a column.

Each of the classes has one matrix, such as the n_i samples of ith class which has a matrix $A_i = [B_{i,1} B_{i,2} \cdots B_{i,n_i}] \in R^{m \times n_i}$. Given a class of training samples and y representing the testing samples of PPIs, the testing sample should be associated with training samples for the given class; y is represented as the linear weighted of the training samples:

$$y = a_{i,1} b_{i,1} + a_{i,2} b_{i,2} + \cdots + a_{i,n_i} b_{i,n_i}. \tag{2}$$

The class of new test sample y is unknown in the prediction of PPIs. When there are a lot of categories, we use the matrix notation and any test sample y is expressed as a linear combination of all the training samples:

$$y = A x_0. \tag{3}$$

x_0 is the weighted matrix of the nonzero weights with the corresponding class; we can determine the class of the new test sample y from x_0:

$$x_0 = \left[0, \ldots, a_{i,1}, a_{i,2}, \ldots, a_{i,n_i}, 0, \ldots, 0\right]^T \in R^n. \tag{4}$$

In order to determine the class that the test sample belongs to, x_0 should be evaluated. From the formula mentioned above, we can see that representation of y is naturally sparse. If y belongs to one class, the nonzero elements in vector x must be associated with that class, and the remaining part is zero which associates with other classes, more categories, and more zeros in vector. The problem can be converted into finding a vector x. In the following optimization problem, $\|x\|_0$ is the l_0-norm of x, and it expresses the number of nonzero elements in vector x:

$$\hat{x}_0 = \arg\min \|x\|_0 \quad \text{subject to } Ax = y. \tag{5}$$

The above problem is an optimization problem with equality constraint. Since the problem is NP-hard problem, in order to solve the problem, (5) can convert to the following l_1-minimization problem:

$$\hat{x}_1 = \arg\min \|x\|_1 \quad \text{subject to } Ax = y. \tag{6}$$

For matrix A, (6) cannot obtain accurate solution, so (6) should be converted to the following generalized version:

$$J(x, \lambda) = \min_x \left\{ \|Ax - y\|_2 + \lambda \|x\|_1 \right\}. \tag{7}$$

Equation (7) is l_1-regularized least square problem that can accept certain extent noise and it is a generalized version of (6). The l_1-regularized least square problem always has a solution. l_1-regularized LS typically yields a sparse vector x that has relatively few nonzero coefficients. Here, $\|x\|_1$ represents the l_1-norm of x and $\lambda > 0$ is the regularization parameter [29]. Through (7), we expect that the classifier can let the output value of the Ax and y as close as possible. The D-value of Ax and y should be as small as possible; also the positive parameter λ in (7) can prevent overfitting. In conclusion, the original problem is showed by sparse representation and then converts to optimization problem (7) by a series of transformations. This optimization problem can be solved by the truncated Newton interior-point method [29].

2.3. Metasample-Based Sparse Representation Classification. The metasamples contain the inherent structural information of training samples. Each subdataset matrix A_i can be factorized into two matrices as follows:

$$A_i \sim B_i C_i. \tag{8}$$

The matrix we used to represent the metasamples from all the k classes after computing the metasamples W_i of each class is as follows:

$$B = \left[B_1, B_2, \ldots, B_k\right]. \tag{9}$$

After converting A into B, SR is computed by minimizing the following equality for a given test sample y:

$$J(x, \lambda) = \min_x \left\{ \|Bx - y\|_2 + \lambda \|x\|_1 \right\}. \tag{10}$$

The optimization problem in (10) is solved using the truncated Newton interior-point method, which is done by l1_ls MATLAB package.

The nonzero entries in the vector x will be all related to the columns of B from a single class i when predicting PPIs without the noise and error; that is to say, the category of the new test sample y is class i. But a few nonzero entries must be related to multiple object classes if the noise and error exists; in order to solve this problem, we use the coefficients from each class to observe how well the test sample can be reconstructed.

The δ_i chooses the coefficients related to the ith class for each class i; it is the feature function. We can reconstruct the given test sample y as $\hat{y}_1 = B\delta_i(x)$, then compute the D-value of y and $\hat{y}$, and finally minimize the D-value as the following equality:

$$\min_i r_i(y) = \|y - B\delta_i(x)\|_2. \tag{11}$$

The flow chart of experiment can be showed in Figure 2.

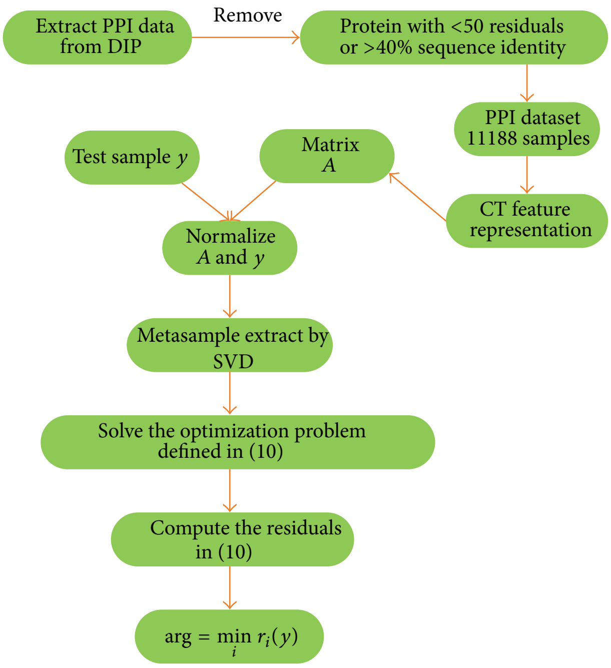
Extract PPI data from DIP
Remove
Protein with <50 residuals or >40% sequence identity
PPI dataset 11188 samples
Test sample y
Matrix A
CT feature representation
Normalize A and y
Metasample extract by SVD
Solve the optimization problem defined in (10)
Compute the residuals in (10)
arg = $\min_i r_i(y)$

Figure 2: The flow chart of classification algorithm.

2.4. Evaluation of Performance. In this paper, accuracy, sensitivity, specificity, and precision were used to measure the performance of the method:

$$\begin{aligned}\text{Accuracy} &= \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FN}+\text{FP}},\\ \text{Sensitivity} &= \frac{\text{TP}}{\text{TP}+\text{FN}},\\ \text{Specificity} &= \frac{\text{TN}}{\text{FP}+\text{TN}},\\ \text{precision} &= \frac{\text{TP}}{\text{FP}+\text{TP}},\end{aligned} \tag{12}$$

where true positive (TP) represents true interaction pair, true negative (TN) represents true noninteraction pair, false positive (FP) represents false interaction pair, and false negative (FN) represents false noninteraction pair. All these indicators are obtained by 5-fold cross validation.

3. Results

3.1. Generation of the Dataset. A dataset of physical protein interactions [34] from Guo et al. [35] has been used during our experiments. We download the database from *S. cerevisiae* core subset of database of interacting proteins (DIP) [36]. There are 5594 protein pairs left to form an eventual positive dataset after removing the protein with fewer than 50 residues or ≥40% sequence identity. The noninteracting pairs comprise another final negative dataset, which were generated from those pairs of proteins that have different subcellular localizations. All these positive datasets and negative datasets come together to form the final dataset that consist of 11188 protein pairs. 80% of the protein pairs from the final dataset were, respectively, randomly used as the training set, and the rest of the protein pairs as the testing set.

Table 1: Division of amino acids based on the dipoles and volumes of the side chains.

Number	Group
1	A, G, V
2	C
3	D, E
4	F, I, L, P
5	H, N, Q, W
6	K, R
7	M, S, T, Y

3.2. Feature Representation. Conjoint triad (CT) [37] is used as feature representation method due to its prediction accuracy in previous study.

CT takes the properties of one amino acid and its vicinal amino acids into account and any three continuous amino acids have been treated as a unit. Therefore, according to the classes of amino acid, we can differentiate the triad. Here, we use a binary space (W, H) to represent a protein sequence; W is the vector of the sequence features; H is the frequency vector corresponding to W. According to the dipoles and volumes of the side chains, the 20 amino acids have been clustered into seven classes, the classification of amino acids is listed in Table 1, and the size of W should be $7 \times 7 \times 7 = 343$. Figure 3 showed the descriptors for (W, H). Eventually, a 686-dimensional vector can be set up to represent each protein pair.

3.3. Classification of PPIs Dataset. The experiment of two-class classification has been completed by the proposed method. Each experiment has been repeated 5 times to acquire the result of high precision. The mean classification accuracies of 5-fold cross validation are charted in Figure 4. Through the experiment, all the accuracy, sensitivity, specificity, and precision can be obtained. Figure 3 shows the classification accuracy on the PPI dataset. In Figure 4, x-axis shows the number of metasamples and y-axis shows the accuracy of classification. As can be seen from Figure 4, it could be drawn that the relationship between the number of metasamples and the accuracy of classification has a general trend of fluctuations. From the graph, it also revealed that the accuracy depends on the number of metasamples. The more the number of metasamples, the higher the accuracy. During the dimension range from 0 to 840, the accuracy is on a steady rise across the board. Then when the count of metasample is 840, the accuracy reaches its highest value about 89.72%. After the number of metasamples dropped below 840, in the area of 840 to 1340, the accuracy begins to decline. In other words, if the number of metasamples is less than 840, the metasample could not be able to capture sufficient inherent structural information of each class. In addition, the training

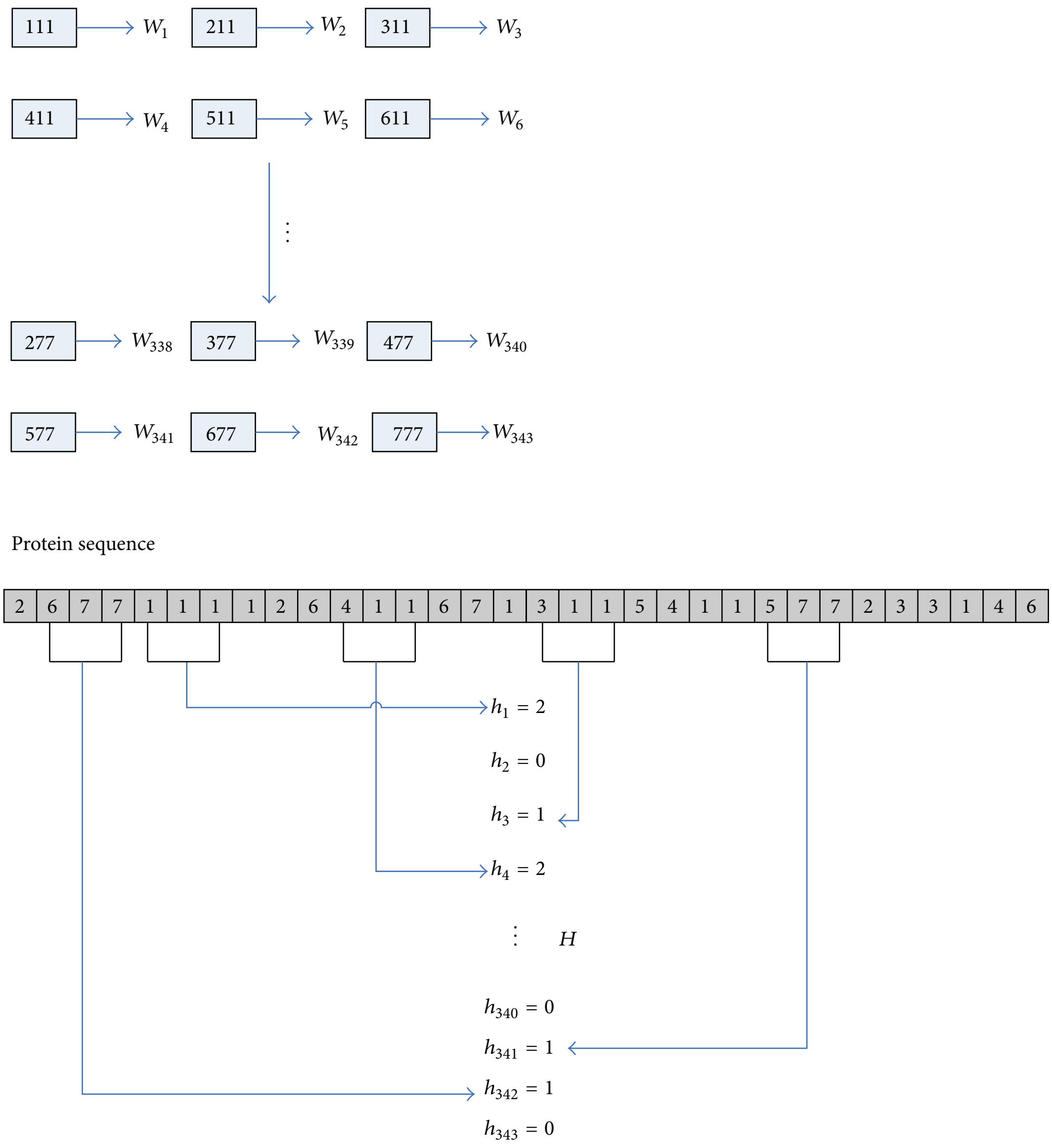

Figure 3: Schematic diagram for constructing the vector space (W, H) of protein sequence.

samples for metasample training cannot be too limited, which is the main weakness of the proposed method.

3.4. Comparison with Other Methods. Among these algorithms, here, the dataset is divided into 80% and 20%, the 80% part representing training set which takes 5-fold cross validation based on SVM (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) to select the optimal parameter of c and g ($c = 8$, $g = 0.001953125$). Then the optimal parameter could apply to the other 20% representing testing set to obtain the result with the accuracy reaching 91.96%. In order to obtain respective accuracy, sensitivity, specificity, and precision, Weka (http://www.cs.waikato.ac.nz/ml/weka/) is used to implement KNN, NN, and NB algorithm. Comparing the performance of MSRC with SVM, KNN, NN, and NB, the result reveals the advantages of MSRC.

Table 2: Comparison of state-of-the-art methods on the PPIs dataset.

	SVM (%)	*K*NN (%)	NN (%)	NB (%)	MSRC (%)
Accuracy	91.96	81.60	63.96	65.47	89.72
Sensitivity	93.11	81.60	64.00	65.5	84.9
Specificity	90.86	79.51	64.14	64.53	94.55
Precision	90.62	81.80	64.00	65.5	93.97

Table 2 shows the accuracy, sensitivity, specificity, and precision in prediction. The result demonstrates that our method is able to correctly predict the PPIs with the accuracy of 89.72%, slightly lower than SVM and obviously higher than KNN, NN, and NB. NN and NB show a distinct worse result in sensitivity than MSRC, with the sensitivity value

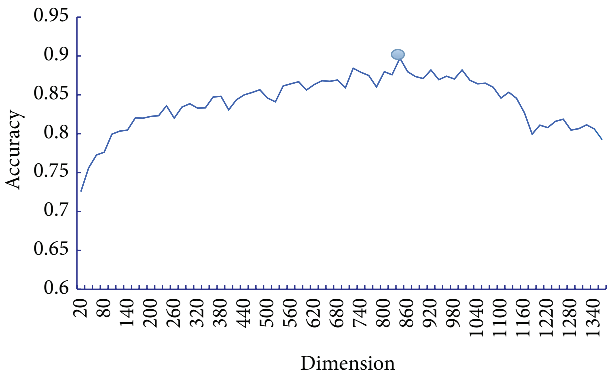

Figure 4: The classification accuracy on the PPIs dataset.

of NN, NB, and MSRC being 64%, 65.5%, and 85.9%, and KNN with the sensitivity of 83.62% is slightly lower than MSRC. SVM has been successfully used for PPIs prediction; considering the characteristics of "high dimensionality" of PPI data, SVM may be the best classifier for predicting PPI data, so the sensitivity value of MSRC is also lower than SVM. In the aspect of specificity, MSRC has the distinct advantage compared to SVM, KNN, NN, and NB, which is at 94.55%. MSRC also does its best in terms of precision, which is 93.97%, much better than other four algorithms.

As can be seen from Table 2 and Figure 4 for the PPIs dataset, MSRC-SVD achieves better classification results than KNN, NN, and NB.

3.5. The Number of Samples for Metasample Training. From the above experimental results, we can see that our method could effectively classify PPI data. The number of metasamples will influence the result of MSRC. The metasamples are extracted by SVD; we should determine the number of metasamples of each class, which is the value of q_i in (8). In the PPIs dataset of this paper, there are only two categories, such that the distinct number of each class is not big. So we make $q_1 = q_2 = q$; the value of q depends on the nested stratified 5-fold cross validation.

In the experiment, SVD is applied to extract metasample from the origin training samples. It extracts data separately aimed at each class. In detail, the method gets samples of equal count from each class to combine the metasamples. Because it should generate eigenvalues and eigenvectors first when reducing the dimension of the SVD matrix, the data from each class in the experiment could emerge as eigenvector of $686 * 686$. Then the eigenvectors corresponding to related rows could be extracted from this eigenvector. In this situation, the number of rows corresponds to the number of samples in each category to be extracted and the data extracted from every row cannot surpass the number of eigenvectors' rows. According to the dataset, the highest number from each class should not surpass 686. As a result, the count of each extracted sample is equal; that is to say, up to a total of 1362 samples can be extracted.

4. Conclusion

PPIs prediction is one of the hot research areas at present. A novel method based on SR was developed for PPIs prediction here. Since the original training samples do not contain the instinct structural information of data as the metasample, MSRC with PPIs uses the SVD to extract a set of metasamples which can represent each testing sample as a linear combination. From the experiment results, we can see that MSRC is efficient in PPIs prediction; the approach can match the better performance than other methods. Moreover, our method is different from other common classification algorithms which construct a model by training samples. In the future, we will investigate how to extract the appropriate number that can improve the accuracy of classification.

Conflict of Interests

All authors declare that they have no competing interests.

Acknowledgments

This work was partially supported by the National Science Foundation Grant (nos. 61203290 and 61175046), the Start-Up Foundation for Doctors of Anhui University (no. 33190078), and outstanding young backbone teachers training (no. 02303301).

References

[1] J.-F. Xia, K. Han, and D.-S. Huang, "Sequence-based prediction of protein-protein interactions by means of rotation forest and autocorrelation descriptor," *Protein & Peptide Letters*, vol. 17, no. 1, pp. 137–145, 2010.

[2] P. Uetz, L. Glot, G. Cagney et al., "A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae*," *Nature*, vol. 403, no. 6770, pp. 623–627, 2000.

[3] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki, "A comprehensive two-hybrid analysis to explore the yeast protein interactome," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 8, pp. 4569–4574, 2001.

[4] A.-C. Gavin, M. Bösche, R. Krause et al., "Functional organization of the yeast proteome by systematic analysis of protein complexes," *Nature*, vol. 415, no. 6868, pp. 141–147, 2002.

[5] H. Zhu, M. Bilgin, R. Bangham et al., "Global analysis of protein activities using proteome chips," *Science*, vol. 293, no. 5537, pp. 2101–2105, 2001.

[6] X. Q. Du, J. X. Cheng, T. T. Zheng, Z. Duan, and F. L. Qian, "A novel feature extraction scheme with ensemble coding for protein-protein interaction prediction," *International Journal of Molecular Sciences*, vol. 15, no. 7, pp. 12731–12749, 2014.

[7] C. H. Liu, K.-C. Li, and S. Yuan, "Human protein-protein interaction prediction by a novel sequence-based co-evolution method: co-evolutionary divergence," *Bioinformatics*, vol. 29, no. 1, pp. 92–98, 2013.

[8] Z.-H. You, Y.-K. Lei, L. Zhu, J. Xia, and B. Wang, "Prediction of protein-protein interactions from amino acid sequences with ensemble extreme learning machines and principal component analysis," *BMC Bioinformatics*, vol. 14, no. 8, article S10, 2013.

[9] J. Zahiri, O. Yaghoubi, M. Mohammad-Noori, R. Ebrahimpour, and A. Masoudi-Nejad, "PPIevo: protein-protein interaction prediction from PSSM based evolutionary information," *Genomics*, vol. 102, no. 4, pp. 237–242, 2013.

[10] Q. C. Zhang, D. Petrey, L. Deng et al., "Structure-based prediction of protein-protein interactions on a genome-wide scale," *Nature*, vol. 490, no. 7421, pp. 556–560, 2012.

[11] S. B. Priya, S. Saha, R. Anishetty, and S. Anishetty, "A matrix based algorithm for protein-protein interaction prediction using domain-domain associations," *Journal of Theoretical Biology*, vol. 326, pp. 36–42, 2013.

[12] J. Planas-Iglesias, J. Bonet, J. García-García, M. A. Marín-López, E. Feliu, and B. Oliva, "Understanding protein-protein interactions using local structural features," *Journal of Molecular Biology*, vol. 425, no. 7, pp. 1210–1224, 2013.

[13] I. Saha, J. Zubek, T. Klingström et al., "Ensemble learning prediction of protein-protein interactions using proteins functional annotations," *Molecular BioSystems*, vol. 10, no. 4, pp. 820–830, 2014.

[14] L. Yang and X. Tang, "Protein-protein interactions prediction based on iterative clique extension with gene ontology filtering," *The Scientific World Journal*, vol. 2014, Article ID 523634, 6 pages, 2014.

[15] O. Souiai, F. Guerfali, S. B. Miled, C. Brun, and A. Benkahla, "In silico prediction of protein–protein interactions in human macrophages," *BMC Research Notes*, vol. 7, article 157, 2014.

[16] S. Martin, D. Roe, and J. L. Faulon, "Predicting protein-protein interactions using signature products," *Bioinformatics*, vol. 21, no. 2, pp. 218–226, 2005.

[17] J. R. Bock and D. A. Gough, "Predicting protein-protein interactions from primary structure," *Bioinformatics*, vol. 17, no. 5, pp. 455–460, 2001.

[18] J. R. Bock and D. A. Gough, "Whole-proteome interaction mining," *Bioinformatics*, vol. 19, no. 1, pp. 125–135, 2003.

[19] K.-C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," *Proteins: Structure, Function and Genetics*, vol. 43, no. 3, pp. 246–255, 2001.

[20] K.-C. Chou, "Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes," *Bioinformatics*, vol. 21, no. 1, pp. 10–19, 2005.

[21] K. C. Chou and Y. D. Cai, "Predicting protein-protein interactions from sequences in a hybridization space," *Journal of Proteome Research*, vol. 5, no. 2, pp. 316–322, 2006.

[22] Y. Guo, L. Yu, Z. Wen, and M. Li, "Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences," *Nucleic Acids Research*, vol. 36, no. 9, pp. 3025–3030, 2008.

[23] S.-W. Zhang, L.-Y. Hao, and T.-H. Zhang, "Prediction of protein-protein interaction with pairwise kernel support vector machine," *International Journal of Molecular Sciences*, vol. 15, no. 2, pp. 3220–3233, 2014.

[24] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[25] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[26] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[27] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[28] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B. Methodological*, vol. 58, no. 1, pp. 267–288, 1996.

[29] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale l1-regularized least squares," *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 606–617, 2007.

[30] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 12, pp. 4164–4169, 2004.

[31] W. Liebermeister, "Linear modes of gene expression determined by independent coponent analysis," *Bioinformatics*, vol. 18, no. 1, pp. 51–60, 2002.

[32] X. Hang and F.-X. Wu, "Sparse representation for classification of tumors using gene expression data," *Journal of Biomedicine and Biotechnology*, vol. 2009, Article ID 403689, 6 pages, 2009.

[33] C.-H. Zheng, L. Zhang, T.-Y. Ng, C. K. Shiu, and D.-S. Huang, "Metasample-based sparse representation for tumor classification," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 5, pp. 1273–1282, 2011.

[34] J.-F. Xia, X.-M. Zhao, and D.-S. Huang, "Predicting protein-protein interactions from protein sequences using meta predictor," *Amino Acids*, vol. 39, no. 5, pp. 1595–1599, 2010.

[35] Y. Z. Guo, L. Z. Yu, Z. N. Wen, and M. L. Li, "Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences," *Nucleic Acids Research*, vol. 36, no. 9, pp. 3025–3030, 2008.

[36] I. Xenarios, Ł. Salwínski, X. J. Duan, P. Higney, S.-M. Kim, and D. Eisenberg, "DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions," *Nucleic Acids Research*, vol. 30, no. 1, pp. 303–305, 2002.

[37] J. Shen, J. Zhang, X. Luo et al., "Predicting protein-protein interactions based only on sequences information," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 11, pp. 4337–4341, 2007.