Predicting Strengths of Protein-protein Interactions Through Online Regression Algorithms

M. Hayashida[1], **M. Kamada**[2], **and H. Koyano**[3]
[1]Department of Electrical Engineering and Computer Science, National Institute of Technology, Matsue College, Matsue, Shimane, Japan
[2]Graduate School of Medicine, Kyoto University, Kyoto, Kyoto, Japan
[3]Quantitative Biology Center, Riken, Kobe, Hyogo, Japan

Abstract— *Protein-protein interactions take various important roles in living cells. Interaction strengths provide useful knowledge to understand complicated cellular networks, and several prediction methods have been developed. In our previous study, we proposed new feature space mappings based on protein domains, and employed support vector regression and relevance vector machine. The combination of the mapping and the supervised regression method outperformed the existing methods.*

In this study, we examine online learning algorithms, the regression passive-aggressive (PA) and adaptive regularization of weights for regression with covariance reset (ARCOR) algorithms. Furthermore, we introduce nonlinear transformation to a linear regression formula and examine ensemble learning. For evaluation, we performed three-fold cross-validation computational experiments. The root mean square error (RMSE) by our proposed method was smaller than those by the existing methods. It implies that our method combining online regression algorithms with nonlinear transformation and sequences of domain regions is useful.

Keywords: protein-protein interaction strength, regression PA, ARCOR

1. Introduction

For understanding dynamic cellular systems, it is necessary to analyze strengths of protein-protein interactions (PPIs). If a protein strongly interacts with another protein, the proteins form a stable complex, and the complex maintains its function. In contrast, if a protein weakly interacts, the protein temporarily changes the state of its complex. In addition, weak interactions are known to be involved with enzyme regulation and signal transduction [1], [2].

The strength of a PPI is often represented by the dissociation constant that is the ratio of the rate constant of the dissociation reaction to that of association reaction. Physicochemical methods for measuring strengths of PPIs have been developed by utilizing solution nuclear magnetic resonance (NMR) [3], [4]. These methods, however, take long time to exhaustively measure whole pairs of proteins in an organism. Hence, several computational methods for predicting strengths of PPIs have been developed. Deng *et al.* proposed a probabilistic model representing protein-protein interactions using domain-domain interactions [5]. LPNM was developed as a linear programming-based method based on the probabilistic model, and minimizes the sum of errors between predicted and actual strengths [6]. ASNM is a faster method developed by modifying the association method [7], the prediction accuracy was comparable to that of LPNM [8]. Chen *et al.* proposed association probabilistic method (APM) by improving ASNM in consideration of the probabilistic model [9]. In our previous study [10], we proposed new feature space mappings from protein pairs using domain information such as amino acid sequences and domain compositions in a protein, and combined them with machine learning methods, support vector regression (SVR) [11] and relevance vector machine (RVM) [12]. The results of the cross-validation experiments showed that the root mean square error (RMSE) by our previous method was smaller than those by the existing methods, APM, ASNM, and LPNM.

Online linear classifiers are fast, reduce memory usage, and have been applied to analyses of big data. In online learning, a learning algorithm takes instances in a sequential manner, and outputs a new model after each observation. Crammer *et al.* proposed a passive-aggressive (PA) algorithm and two alternative modifications for coping with noise [13]. The PA algorithms update weights as little as possible such that the current training instance is correctly classified. Adaptive regularization of weights (AROW) is another online algorithm, can deal with non-separable data, and achieves state-of-the-art performance [14]. These classification algorithms can be extended to linear regression problems. The ARCOR algorithm is a modification of AROW for regression [15]. We examine the regression PA and ARCOR algorithms for improving prediction accuracy of strengths of PPIs. Furthermore, we introduce nonlinear transformation to a linear regression formula and examine ensemble learning. For evaluation, we performed three-fold cross-validation computational experiments. The RMSE by our proposed method was smaller than those by the existing methods.

ISBN: 1-60132-466-9, CSREA Press ©

2. Methods

In this section, we briefly review our previously proposed feature space mapping, called SPD [10], and online learning algorithms, the regression passive-aggressive (PA) algorithm [13] and adaptive regularization of weights for regression with covariance reset (ARCOR) [15]. We explain the proposed method using nonlinear transformation and ensemble learning.

2.1 Feature space mapping by restriction of spectrum kernel to domain regions (SPD)

The feature space mapping of SPD is obtained by restricting the application of the spectrum kernel [16] to domain regions (see Fig. 1).

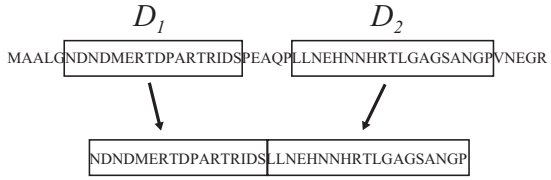

Fig. 1: Illustration on the feature space mapping of restriction of spectrum kernel to domain regions (SPD). A protein sequence contains two domains D_1 and D_2. Spectrum kernel is applied to the concatenated amino acid string in domain regions.

Let $\mathcal{A}$ and $\mathcal{A}^k$ be an alphabet and the set of all strings with length k consisting of letters in $\mathcal{A}$, respectively. Let $\phi_s(P_i)$ be the number of occurrences of string s in the sequence restricted to the domain regions in protein P_i. The feature space mapping $\boldsymbol{f}_{ij}$ of SPD for a pair of proteins P_i and P_j is defined by

$$f_{ij}^{(n)} = \phi_{s_n}(P_i), \quad f_{ij}^{(|\mathcal{A}|^k+n)} = \phi_{s_n}(P_j) \tag{1}$$

for all $s_n \in \mathcal{A}^k$, where $|\mathcal{A}|$ indicates the number of elements in $\mathcal{A}$.

2.2 Passive-aggressive (PA) algorithm

Let $(\boldsymbol{x}_t, y_t)$ be t-th example, where $\boldsymbol{x}_t$ and y_t mean a feature vector obtained from a pair of proteins P_i and P_j and the strength that P_i and P_j interact with each other for our purpose. Then, online linear learners find a weight vector $\boldsymbol{w}$ such that $\boldsymbol{w} \cdot \boldsymbol{x}_t$ is close to y_t. Let $\boldsymbol{w}_t$ be the weight vector on round t. $\boldsymbol{w}_1$ is initialized to $(0, \ldots, 0)$. In the regression passive-aggressive algorithm, the new weight $\boldsymbol{w}_{t+1}$ is determined depending on the t-th example $(\boldsymbol{x}_t, y_t)$ such that it minimizes $||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||^2$ under the constraint $l(\boldsymbol{w}_{t+1}; (\boldsymbol{x}_t, y_t)) = \max\{0, |y_t - \boldsymbol{w}_{t+1} \cdot \boldsymbol{x}_t| - \epsilon\} = 0$ for some $\epsilon > 0$. If $l(\boldsymbol{w}_t; (\boldsymbol{x}_t, y_t)) = l_t = 0$, then $\boldsymbol{w}_{t+1}$ is determined to be $\boldsymbol{w}_t$ to minimize $||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||^2$. Otherwise, $\boldsymbol{w}_{t+1}$ is aggressively determined to satisfy $l(\boldsymbol{w}_{t+1}; (\boldsymbol{x}_t, y_t)) = 0$. The Lagrange function is defined as $\mathcal{L}(\boldsymbol{w}_{t+1}, \tau) = ||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||^2 + \tau(|y_t - \boldsymbol{w}_{t+1} \cdot \boldsymbol{x}_t| - \epsilon)$ using a Lagrange multiplier $\tau > 0$. By solving $\partial\mathcal{L}(\boldsymbol{w}_{t+1}, \tau)/\partial \boldsymbol{w}_{t+1} = 0$ and $\partial\mathcal{L}(\boldsymbol{w}_{t+1}, \tau)/\partial\tau = 0$, we have the update formula,

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \text{sign}(y_t - \boldsymbol{w}_t \cdot \boldsymbol{x}_t)\tau_t \boldsymbol{x}_t, \tag{2}$$

where $\tau_t = \frac{l_t}{||\boldsymbol{x}_t||^2}$, and $\text{sign}(z) = -1$ if $z < 0$, 0 if $z = 0$, 1 if $z > 0$.

By allowing $l(\boldsymbol{w}_{t+1}; (\boldsymbol{x}_t, y_t)) \neq 0$ and introducing a constant $C > 0$, called aggressiveness parameter, two variants were proposed. One is to minimize $||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||^2 + C\xi$ under the constraints $l(\boldsymbol{w}_{t+1}; (\boldsymbol{x}_t, y_t)) \leq \xi$ and $\xi \geq 0$. Another is to minimize $||\boldsymbol{w}_{t+1} - \boldsymbol{w}_t||^2 + C\xi^2$ under the constraint $l(\boldsymbol{w}_{t+1}; (\boldsymbol{x}_t, y_t)) \leq \xi$. Then, the corresponding update formulas are given by replacing τ_t in Eq. (2) with $\min\{C, \frac{l_t}{||\boldsymbol{x}_t||^2}\}$ for PA-I, and with $\frac{l_t}{||\boldsymbol{x}_t||^2 + \frac{1}{2C}}$ for PA-II, respectively.

2.3 Adaptive regularization of weights for regression with covariance reset (ARCOR)

In the ARCOR algorithm, a Gaussian distribution $\mathcal{N}(\boldsymbol{w}_t, \Sigma_t)$ with a mean vector $\boldsymbol{w}_t$ and a covariance matrix Σ_t is maintained. First, $\boldsymbol{w}_1$ and Σ_1 are initialized as $(0, \cdots, 0)$ and the identity matrix, respectively. Given the t-th example $(\boldsymbol{x}_t, y_t)$, the new weight $\boldsymbol{w}_{t+1}$ and covariance matrix Σ_{t+1} are determined such that they minimize $D_{KL}(\mathcal{N}(\boldsymbol{w}_{t+1}, \Sigma_{t+1})||\mathcal{N}(\boldsymbol{w}_t, \Sigma_t)) + \frac{1}{2r}||y_t - \boldsymbol{w}_{t+1} \cdot \boldsymbol{x}_t||^2 + \frac{1}{2r}\boldsymbol{x}_t^\top \Sigma_{t+1} \boldsymbol{x}_t$, where D_{KL} indicates the Kullback-Leibler divergence, r is a positive constant, and $\boldsymbol{x}^\top$ indicates the transpose of $\boldsymbol{x}$. The first term of the objective function requires not much to change the parameters $\boldsymbol{w}_{t+1}$ and Σ_{t+1} from $\boldsymbol{w}_t$ and Σ_t. The second term requires to minimize the squared error for the current example, and the last term requires to reduce the variance for the parameter $\boldsymbol{w}_{t+1}$. Then, the update formula is derived from the minimization problem as $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + (y_t - \boldsymbol{w}_t \cdot \boldsymbol{x}_t)\Sigma_t \boldsymbol{x}_t / (r + \boldsymbol{x}_t^\top \Sigma_t \boldsymbol{x}_t)$ and $\Sigma_{t+1}^{-1} = \Sigma_t^{-1} + \frac{1}{r}\boldsymbol{x}_t \boldsymbol{x}_t^\top$.

2.4 Nonlinear transformation and ensemble learning

In linear regression algorithms including PA-I, PA-II, and ARCOR, the weight $\boldsymbol{w}$ of $y = \boldsymbol{w} \cdot \boldsymbol{x}$ is often estimated such that the value of some cost function for all examples $(\boldsymbol{x}_t, y_t)$ is close to zero. However, actual values y_t do not always follow such a linear formula, $\boldsymbol{w} \cdot \boldsymbol{x}$. Especially, for our purpose, in a feature vector $\boldsymbol{f}_{ij}$ for a pair of proteins P_i and P_j, a part of the feature vector involved with P_i can be related to those with P_j. Hence, it is reasonable that quadratic or more terms of $x_i x_j$ and $x_i x_j x_k$ are included in the predictive function as well as linear terms. Thus, we introduce a nonlinear transformation function g as $y = g(\boldsymbol{w} \cdot \boldsymbol{x})$.

In addition, we examine a simple ensemble learning using the weights $\boldsymbol{w}^{(m)}$ $(m = 1, \ldots, M)$ obtained by M

regression learners, that is, we predict the interaction strength between proteins P_i and P_j as $\frac{1}{M}\sum_{m=1}^{M} g(\boldsymbol{w}^{(m)} \cdot \boldsymbol{f}_{ij})$.

3. Results

In the previous studies [6], [9], interaction sequence tags (ISTs) obtained by high-throughput yeast two-hybrid (Y2H) experiments [17] were used as strengths of protein-protein interactions. It, however, is known that Y2H includes a high false-positive rate mainly caused by non-specific interactions [18]. Hence, we used more reliable WI-PHI protein-protein interaction dataset [19] with 50000 protein pairs, in which PPIs are weighted by some reliability calculated in a statistical manner from several biological experiments, as interaction strengths under the difficulty of measuring strengths for many protein pairs. In our preliminary study [20], it was shown also for the IST dataset that our previous method 'SVR+SPD' outperformed the best existing method APM. We used the value dividing the weight of PPI by the maximum weight as the strength. We calculated the SPD feature vector using amino acid sequences and domain compositions of proteins stored in UniProt database [21]. Among 50000 protein pairs, we used 1387 pairs that contain complete domain sequences, which include 758 proteins and 327 domains. We added 100 randomly selected protein pairs as PPIs with strength 0. Totally 1487 protein pairs are the same as those in the previous experiment. The alphabet $\mathcal{A}$ consists of 20 amino acids and ambiguous one. For evaluating prediction accuracy, we performed three-fold cross-validation experiments, and calculated RMSE defined by $\sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2}$ for N actual values y_t and predicted values $\hat{y}_t$. We used the regression PA and ARCOR algorithms implemented in Hivemall with default parameters [22], which run under Hadoop scalable and distributed computing environment [23].

Table 1 shows the results of RMSE for test data by our proposed method using SPD with $k = 4$, our previous methods, and APM. 'SPD+PA-I' indicates the combination of SPD and PA-I. 'ARCOR-H' indicates the algorithm obtained by replacing the loss function of ARCOR with the hinge loss function as $l(\boldsymbol{w}; (\boldsymbol{x}_t, y_t)) = \max\{0, |y_t - \boldsymbol{w} \cdot \boldsymbol{x}_t| - \epsilon\}$. 'ARCOR-H$\sigma$', 'PA-I$\sigma$' and 'PA-II$\sigma$' indicate the algorithms obtained by replacing ϵ of the hinge loss function with $\epsilon\sigma_t$, respectively, where σ_t is the standard deviation of $y_1, \ldots, y_t$. 'without' and 'with bias' mean whether or not a bias term b is added as $y = g(\boldsymbol{w} \cdot \boldsymbol{x} + b)$. In our previous methods, 'RVM+SPD', 'SVR+SPD', 'RVM+DN', and 'SVR+DN', the Laplacian kernel $K(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\sigma||\boldsymbol{x} - \boldsymbol{y}||)$ was employed instead of the dot product. 'DN' indicates a feature vector concerning the number of domains in a protein. For the case of $g(y) = y$, the RMSE by our proposed method was larger than that by 'RVM+SPD'. For the case of $g(y) = y^{\frac{1}{2}}$, the RMSE by the ensemble method with a bias term was the smallest in the experiments. It implies that the nonlinear transformation is useful for predicting the strength of PPIs.

4. Conclusion

We proposed a simple ensemble learning with nonlinear transformation for predicting strengths of protein-protein interactions. For evaluation of our proposed method, we performed three-fold cross-validation experiments. The RMSE by our proposed method was smaller than those by the existing methods, APM, and our previously developed methods. It implies that our method combining online regression algorithms with nonlinear transformation and sequences of domain regions is useful although further evaluation for more data directly related to the dissociation constant is required. In this study, we used online learners, the regression PA and ARCOR algorithms for linear regression. These algorithms, however, can also handle kernel functions. For instance, the PA-I algorithm was combined with the polynomial kernel and applied to dependency parsing and hyponymy-relation extraction [24]. For further improvement of prediction accuracy, we would like to employ kernel functions combined with the regression PA and ARCOR algorithms.

Acknowledgements

This work was partially supported by Grants-in-Aid #16K00392, and #16KT0020 from MEXT, Japan.

References

[1] I. Nooren and J. Thornton, "Diversity of protein-protein interactions," *EMBO Journal*, vol. 22, pp. 3486–3492, 2003.

[2] J. Vaynberg and J. Qin, "Weak protein-protein interactions as probed by NMR spectroscopy," *Trends in Biotechnology*, vol. 24, pp. 22–27, 2006.

[3] P. Kastritis and A. Bonvin, "On the binding affinity of macromolecular interactions: daring to ask why proteins interact," *Journal of the Royal Society Interface*, vol. 10, p. 20120835, 2013.

[4] Z. Liu, Z. Gong, X. Dong, and C. Tang, "Transient protein-protein interactions visualized by solution NMR," *Biochimica et Biophysica Acta*, vol. 1864, pp. 115–122, 2016.

[5] M. Deng, S. Mehta, F. Sun, and T. Chen, "Inferring domain-domain interactions from protein-protein interactions," *Genome Research*, vol. 12, pp. 1540–1548, 2002.

[6] M. Hayashida, N. Ueda, and T. Akutsu, "Inferring strengths of protein-protein interactions from experimental data using linear programming," *Bioinformatics*, vol. 19, pp. ii58–ii65, 2003.

[7] E. Sprinzak and H. Margalit, "Correlated sequence-signatures as markets of protein-protein interaction," *Journal of Molecular Biology*, vol. 311, pp. 681–692, 2001.

[8] M. Hayashida, N. Ueda, and T. Akutsu, "A simple method for inferring strengths of protein-protein interactions," *Genome Informatics*, vol. 15, pp. 56–68, 2004.

[9] L. Chen, L.-Y. Wu, Y. Wang, and X.-S. Zhang, "Inferring protein interactions from experimental data by association probabilistic method," *Proteins: Structure, Function, and Bioinformatics*, vol. 62, pp. 833–837, 2006.

[10] M. Kamada, Y. Sakuma, M. Hayashida, and T. Akutsu, "Prediction of protein-protein interaction strength using domain features with supervised regression," *The Scientific World Journal*, vol. 2014, p. 240673, 2014.

[11] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer, 1995.

Table 1: Result of root mean square error (RMSE) for test data by our proposed method using SPD with $k = 4$, and related methods.

	$g(y) = y$		$y^{\frac{1}{2}}$	
method	without	with bias	without	with bias
SPD+PA-I	0.1539	0.1451	0.1288	0.1304
SPD+PA-Iσ	0.1539	0.1451	0.1288	0.1304
SPD+PA-II	0.1547	0.1457	0.1274	0.1215
SPD+PA-IIσ	0.1548	0.1458	0.1275	0.1215
SPD+ARCOR	0.1558	0.1476	0.1280	0.1221
SPD+ARCOR-H	0.1538	0.1454	0.1288	0.1303
SPD+ARCOR-Hσ	0.1546	0.1462	0.1274	0.1217
Ensemble	0.1544	0.1458	0.1235	**0.1195**
RVM+SPD(k=2)[10]		0.12470		
SVR+SPD(k=2)[10]		0.12654		
RVM+DN[10]		0.12873		
SVR+DN[10]		0.12573		
RVM+APM[10]		0.13556		
SVR+APM[10]		0.13112		
APM[9]		0.13517		

[12] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[13] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *Journal of Machine Learning Research*, vol. 7, pp. 551–585, 2006.

[14] K. Crammer, A. Kulesza, and M. Dredze, "Adaptive regularization of weight vectors," *Advances in Neural Information Processing Systems*, vol. 23, pp. 414–422, 2009.

[15] N. Vaits and K. Crammer, "Re-adapting the regularization of weights for non-stationary regression," in *The 22nd International Conference on Algorithmic Learning Theory*, 2011.

[16] C. Leslie, E. Eskin, and W. Noble, "The spectrum kernel: a string kernel for SVM protein classification," in *Proceedings of Pacific Symposium on Biocomputing 2002*, 2002, pp. 564–575.

[17] 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 USA*, vol. 98, pp. 4569–4574, 2001.

[18] A. Brückner, C. Polge, N. Lentze, D. Auerbach, and U. Schlattner, "Yeast two-hybrid, a powerful tool for systems biology," *International Journal of Molecular Sciences*, vol. 10, pp. 2763–2788, 2009.

[19] L. Kiemer, S. Costa, M. Ueffing, and G. Cesareni, "WI-PHI: A weighted yeast interactome enriched for direct physical interactions," *Proteomics*, vol. 7, pp. 932–943, 2007.

[20] Y. Sakuma, M. Kamada, M. Hayashida, and T. Akutsu, "Inferring strengths of protein-protein interactions using support vector regression," in *The 19th International Conference on Parallel and Distributed Processing Techniques and Applications*, 2013.

[21] The UniProt Consortium, "Reorganizing the protein space at the Universal Protein Resource (UniProt)," *Nucleic Acids Research*, vol. 40, pp. D71–D75, 2012.

[22] M. Yui and I. Kojima, "A database-hadoop hybrid approach to scalable machine learning," in *2013 IEEE International Congress on BigData*, 2013.

[23] https://hadoop.apache.org/.

[24] N. Yoshinaga and M. Kitsuregawa, "Kernel slicing: scalable online training with conjunctive features," in *The 23th International Conference on Computational Linguistics*, 2010, pp. 1245–1253.