Biochemical and Biophysical Research Communications 400 (2010) 219–224

Contents lists available at ScienceDirect

Biochemical and Biophysical Research Communications

journal homepage: www.elsevier.com/locate/ybbrc

Application of maximin correlation analysis to classifying protein environments for function prediction

Taehoon Lee [a,1], Hyeyoung Min [b,1], Seung Jean Kim [c], Sungroh Yoon [a,*]

[a] School of Electrical Engineering, Korea University, Seoul 136-713, Republic of Korea
[b] College of Pharmacy, Chung-Ang University, Seoul 156-756, Republic of Korea
[c] Department of Electrical Engineering, Stanford University, CA 94305, USA

ARTICLE INFO

Article history:
Received 8 August 2010
Available online 16 August 2010

Keywords:
Bioinformatics
Pattern recognition
Supervised classification
Maximin correlation analysis
Protein function prediction

ABSTRACT

More and more protein structures are being discovered, but most of these still have little functional information. Based on the assumption that structural resemblance would lead to functional similarity, researchers computationally compare a new structure with functionally annotated structures, for high-throughput function prediction. The effectiveness of this approach depends critically upon the quality of comparison. In particular, robust classification often becomes difficult when a function class is an aggregate of multiple subclasses, as is the case with protein annotations. For such multiple-subclass classification problems, an optimal method termed the maximin correlation analysis (MCA) was proposed. However, MCA has never been applied to automated protein function prediction although MCA can minimize the misclassification risk in the correlation-based nearest neighbor classification, thus increasing classification accuracy. In this article, we apply MCA to classifying three-dimensional protein local environment data derived from a subset of the protein data bank (PDB). In our framework, the MCA-based classifier outperformed the compared alternatives by 7–19% and 6–27% in terms of average sensitivity and specificity, respectively. Given that correlation-based similarity measures have been widely used for mining protein data, we expect that MCA would be employed to enhance other types of automated function prediction methods.

1. Introduction

Owing to recent initiatives in structural genomics, a great number of protein three-dimensional structures as well as two-dimensional information have been stored in the protein data bank (PDB) [1,2]. However, protein function annotations remain mostly unknown, motivating the needs for developing automated function prediction methods and for making the structure data available for the study of biological systems. Accordingly, many computational methods have been proposed as a predictive tool of protein functions.

Existing approaches can be classified into three types: sequence-based, structure-based and hybrid. First, sequence-based methods transfer annotations from characterized proteins to a homologue with unknown function [3]. These methods mostly exploit the power of alignment and clustering [4], and extra features such as evolutionary traces [5] and gene ontology annotations [6] are often employed. Second, structure-based techniques mainly utilize structure information and are often more reliable than sequence-based tools, since structures are better preserved than sequences, and proteins with little or no sequence similarity can also have common structures [6,7]. Examples include PHUNCTIONER [7], FLORA [8] and ConSurf [9], which are based on gene ontology, structural pattern recognition, and evolutionary conservation, respectively. Additionally, Landgraf et al. performed 3-D cluster analysis through global and regional alignments [10], whereas Hvidsten et al. established the structure–function relationship based only on the local substructure similarity [11]. Lastly, hybrid function prediction methods such as GOdot [12] and ProFunc [13] integrate both sequence and structure similarity for function prediction.

At the core of many automated function predictors is supervised classification [14], which typically consists of training and prediction phases. In the training phase, labeled examples are prepared for training a classifier. Proteins annotated with an identical function are grouped into a class. For each class, the classifier is thus trained using multiple instances which belong to the class.

Abbreviations: KNN, *k*-nearest neighbor; MCA, maximin correlation analysis; SVM, support vector machine.

* Corresponding author. Address: Room 207, Engineering Bldg., School of Electrical Engineering, Korea University, Anamdong, Seongbukgu, Seoul 136-713, Republic of Korea. Fax: +82 2 3290 3844.
E-mail address: sryoon@korea.ac.kr (S. Yoon).

[1] Contributed equally.

0006-291X/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.bbrc.2010.08.042

Conceptually, many classifiers work by internally constructing a *template* (or model) of each class that can compactly represent all the instances belonging to the class [14–16]. In the prediction phase, a protein structure with an unknown function is compared with each class template by the trained classifier. After identifying the class whose template matches the new structure most closely in terms of the criteria used, the classifier predicts the function of this new protein as that of the identified class.

Inherently, some protein functions are closely related, and the corresponding classes can further be grouped into an aggregate class. For instance, histones H2A, H2B, H3 and H4 are all related in living organisms and can be collected under an aggregate class 'Histone.' This hierarchical class organization is often better than the flat organization in that the former resembles the biological hierarchy being modeled more closely. For such multiple-subclass classification problems, a robust method termed the *maximin correlation analysis* (MCA) was proposed [17]. MCA can find the optimal aggregate template when correlation is used as the similarity measure. This aggregate template is called the *maximin (correlation) aggregate template* in that it maximizes the minimum correlation with the templates it represents, thus minimizing the maximum misclassification risk in the correlation-based nearest neighbor classification.

Despite this desirable property, MCA has not been widely known to the bioinformatics community and thus has never been applied to automated protein function prediction. In this work, we aim to elucidate how effective MCA can be for improving classification-based automated protein function prediction. To this end, we perform MCA on three-dimensional protein structure data represented as FEATURE vectors [18–20], which characterize local environments around residues by counting physicochemical properties within concentric shells around a central point on a residue. In this FEATURE framework, we compare the proposed MCA-based classifier with alternative classification techniques in terms of specificity and sensitivity.

2. Materials and methods

2.1. Maximin correlation analysis (MCA)

MCA arises in correlation-based pattern recognition, in which the correlation between two vectors is called the *cosine similarity metric* [16]; we use the two terms interchangeably in this article. Note that the correlation defined in the current context is different from the popular Pearson's correlation coefficient [21] although they are connected in that the cosine similarity of a mean-centered, deviation-normalized data is the same as the Pearson's correlation coefficient of the data.

The correlation between two nonzero vectors x and y in $\mathbb{R}^{\mathbf{n}}$ is defined as

$$\phi(x,y)=\frac{x^Ty}{\|x\|\|y\|}. \tag{1}$$

A basic property of this correlation function is that it is symmetric, *i.e.*, $\phi(x,y)=\phi(y,x)$. Another property is that it is (positive) homogeneous (of degree 0) in x for fixed y and vice versa: For all $t>0$,

$$\phi(tx,y)=\phi(x,y). \tag{2}$$

Finally, the correlation $\phi(x,y)$ is quasi-concave in x for fixed y (and vice versa) when it is positive: For $\gamma\geqslant 0$, the set

$$\{x|\phi(x,y)\geqslant\gamma\}=\{x|\gamma\|x\|\|y\|\leqslant x^Ty\} \tag{3}$$

is convex (since it is a second-order cone in $\mathbb{R}^{\mathbf{n}}$ [22]. In this study, the correlation $\phi(x,y)$ is used to measure the similarity between two FEATURE vectors x and y.

For a given nonzero vector x and a non-empty set $\mathcal{Y}\subseteq\mathbb{R}^{\mathbf{n}}$, the minimum correlation between x and $\mathcal{Y}\subseteq\mathbb{R}^{\mathbf{n}}\setminus\{\mathbf{0}\}$ is

$$\phi(x,\mathcal{Y})=\inf_{y\in\mathcal{Y}}\phi(x,y) \tag{4}$$

When it is positive, $\phi(x,\mathcal{Y})$ is quasi-concave in x for fixed $\mathcal{Y}$. This property can be seen from the fact that the worst-case correlation is the infimum of quasi-concave functions and quasi-concavity is preserved under the operation of infimum [22]. In the current setup, $\mathcal{Y}$ corresponds to a protein class that contains a number of FEATURE vectors as members.

We are interested in the problem of finding a nonzero vector $\mathcal{Y}\subseteq\mathbb{R}^{\mathbf{n}}$ that maximizes the minimum correlation with the set $\mathcal{Y}$:

$$\begin{aligned}&\text{Maximize }\phi(x,\mathcal{Y})\\&\text{subject to }x\neq 0.\end{aligned} \tag{5}$$

This problem is called the *maximin correlation analysis problem* (MCAP) (with the set $\mathcal{Y}$). From the homogeneous property of the correlation, we can see that this problem is positive homogeneous, meaning that if x is a solution, then for any $t>0$, tx is a solution. Moreover, it is unique up to positive scaling [17].

The MCAP has a simple geometric interpretation via the following: The arccosine of the correlation between two nonzero vectors x and y in $\mathbb{R}^{\mathbf{n}}$ is the angle between the two vectors, namely

$$\angle(x,y)=\cos^{-1}\left(\frac{x^Ty}{\|x\|\|y\|}\right).$$

It is then trivial to see that the problem of finding a nonzero vector $x\in\mathbb{R}^{\mathbf{n}}$ that minimizes the worst-case angle is equivalent to the MCAP. Fig. 1A illustrates the worst-case angle minimization problem with an ellipse in $\mathbb{R}^{\mathbf{2}}$. Here, the angular center line does not pass through the centroid $\bar{y}$ of the ellipse $\mathcal{Y}$. This angular center line corresponds to the *maximin template* of the group represented by $\mathcal{Y}$.

To summarize, identifying the maximin template of a protein class $\mathcal{Y}$ corresponds to finding the vector that can represent the whole $\mathcal{Y}$ in such a way that the minimum correlation (*i.e.*, the worst-case similarity) between this vector and the members of class $\mathcal{Y}$ is maximized. Note that other types of templates such as the centroid and the medoid templates (*i.e.*, mean and median, respectively) cannot in general represent $\mathcal{Y}$ in this way. Since the minimum similarity is maximized by representing a class by its maximin template, we can minimize the risk of misclassification in the correlation-based nearest neighbor classification by the maximin templates of classes.

2.2. The FEATURE framework

We apply MCA to FEATURE [18–20], a versatile framework that can be used for modeling and recognizing functional sites in macromolecular structures. FEATURE vectors can characterize local three-dimensional environments around protein residues by counting physicochemical properties within co-centric shells around a point on each residue. Examples of such properties include aliphatic carbon, aromatic carbon and nitrogen, amide carbon and oxygen, carboxyl and hydroxyl oxygen, sulfur, Van der Waals volume, (partial) charge, hydrophobicity and solvent accessibility; refer to [20] for more details. The measurement shells are centered at a common point on a residue, and their radii are multiples of a user-specified value.

2.3. Data preparation and preprocessing

We derived 1,992,567 FEATURE vectors from a subset of PDB according to the procedure described in [20]. To remove redundancy, this subset was prepared in such a way that no two

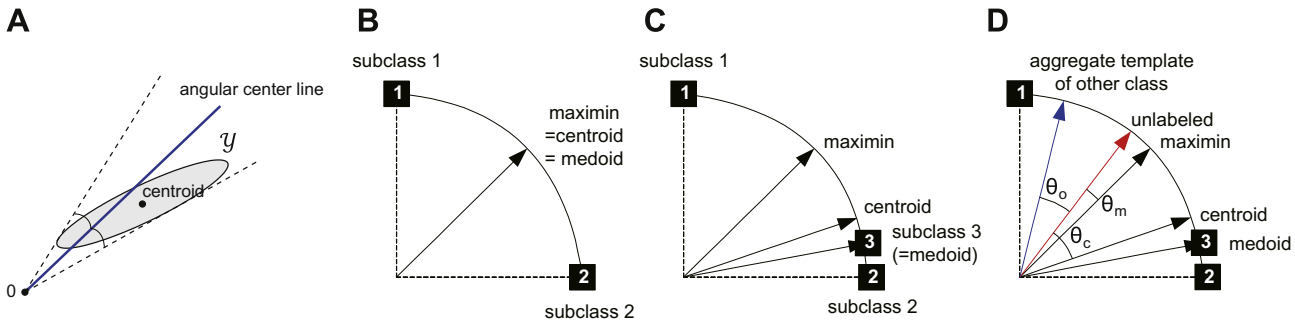

Fig. 1. Geometric interpretation of the maximin aggregate template. (A) The angular center line of an ellipse. The angular center line does not pass through the centroid $\bar{y}$ (*i.e.*, the average) of the ellipse y. (B–D) An example to explain why using the maximin aggregate template would produce a better classification result than using the conventional centroid or medoid template: (B) When only two subclasses exist, the three types of aggregate templates are all aligned. (C) If there exists another subclass (*i.e.*, subclass 3 in the figure), and its template is located between those of subclasses 1 and 2, then the three aggregate templates all become different. The maximin aggregate template does not move, but the location of the centroid template changes. By definition, the medoid aggregate template corresponds to the template of subclass 3. (D) If $\theta_m < \theta_o < \theta_c$, as indicated in the figure, then using the centroid aggregate template (as well as the medoid aggregate template, whose angle with the unlabeled object is even greater than θ_c) would result in misclassification of the unlabeled vector, whereas using the maximin aggregate template would not.

structures have greater than 50% sequence similarity. Each vector represents the 44 physicochemical properties listed in [20], where each property is measured along 6 co-centric shells with radii of multiples of 1.25 Angstroms (*i.e.*, the innermost shell has the radius of 1.25 Angstroms, the second has 2.5 Angstroms and so on). The total number of dimensions in a vector is thus 44×6 or 264.

We further identified those FEATURE vectors that have PROSITE [23–25] annotations and grouped these vectors with respect to their PROSITE annotations. It would be ideal to classify these vectors according to their real biological classes and to see how MCA can rediscover these inherent classes. However, many of the structures stored in PDB are not experimentally verified, and we decided to use the annotations in PROSITE, a protein database independent of PDB, in lieu of experimentally verified class labels. Finally, we selected 89 groups that have at least 3 and at most 41 subclasses. The total number of vectors included in these 89 groups was 17,752. More details of the 89 groups and the vectors used are available from http://dna.korea.ac.kr/pub/mca.

2.4. Classifier implementation

A solver that can solve the MCA problem was implemented with MATLAB. For comparison, the following alternative classifiers were considered: a classifier based on the conventional centroid and median templates, the WebFEATURE [19] classifier that employs the naïve Bayes model [14], a support vector machine (SVM) based classifier [15], and the *k*-nearest neighbor (KNN) classifier [16]. The WebFEATURE classifier was downloaded from http://feature.stanford.edu and then customized for our experiments. The SVM-based classifier were implemented on top of WebFEATURE, by replacing its classification engine with the SVM coded using the LIBSVM package [26]. All the other methods used for comparison were implemented with MATLAB.

2.5. Performance measurement

For each of the classifiers used, we tested how well it can distinguish a class from another. For a pair of protein classes, we randomly selected 30% of the FEATURE vectors from one class as positive samples for training and the same number of negative samples from the other class. The instances not selected for training were used as test data for computing sensitivity and specificity. This procedure was repeated for each pair of protein classes, namely $\binom{89}{2} = 3916$ times, for each type of classifier. To assess classification performance, we performed 10-fold cross-validation [14,16] and measured the average sensitivity and specificity. For additional evaluation, a small number of random samples (on average 9.915 samples per class) were taken from each of the 89 classes and were used for training. This sampling process was repeated 10 times, and the mean sensitivity and specificity were computed. Throughout the classification experiments we carried out, the execution time of all compared methods remained reasonable, not exceeding a few tens of minutes, on a typical 2.66-GHz Linux machine with 8-GB memory.

3. Results and discussion

3.1. Verification of the maximin aggregate templates identified

We compared the maximin, centroid and medoid aggregate templates of a class with respect to their minimum correlation with the individual vectors in each subclass of the class, as shown in Fig. 2. The key observation is that the plot for the maximin aggregate template always has the largest minimum value. That is, the worst-case (*i.e.*, minimum) correlation is maximized by using the maximin aggregate template, although it does not always show the highest level of correlation. For instance, in Fig. 2B, the worst-case correlation occurs in subclass 2 for all types of aggregate templates, and the minimum value is the largest for the maximin case. We observed the same phenomena for all the 89 classes tested. This experiment suggests that, by employing the maximin-based approach, we may reduce the degree of misclassification committed by traditional centroid- or medoid-based classification techniques.

3.2. Classification performance comparison

To test the reasoning in the previous paragraph, we performed classification of FEATURE vectors using the maximin aggregate templates of the 89 protein classes. An unlabeled data object was classified into the class whose maximin aggregate template is closest to this object. Fig. 3A shows the performance[2] of this MCA-based classifier measured in terms of sensitivity and specificity [16]. For comparison, we also tried classification by using the alternative classification techniques described in Section 2. Note that, for

[2] For some algorithms used in comparison, it is difficult to define a discrimination threshold that can be varied to draw a receiver operating characteristic (ROC) curve [16]. We thus show the distribution of sensitivity and specificity over every classification instance, rather than presenting ROC curves.

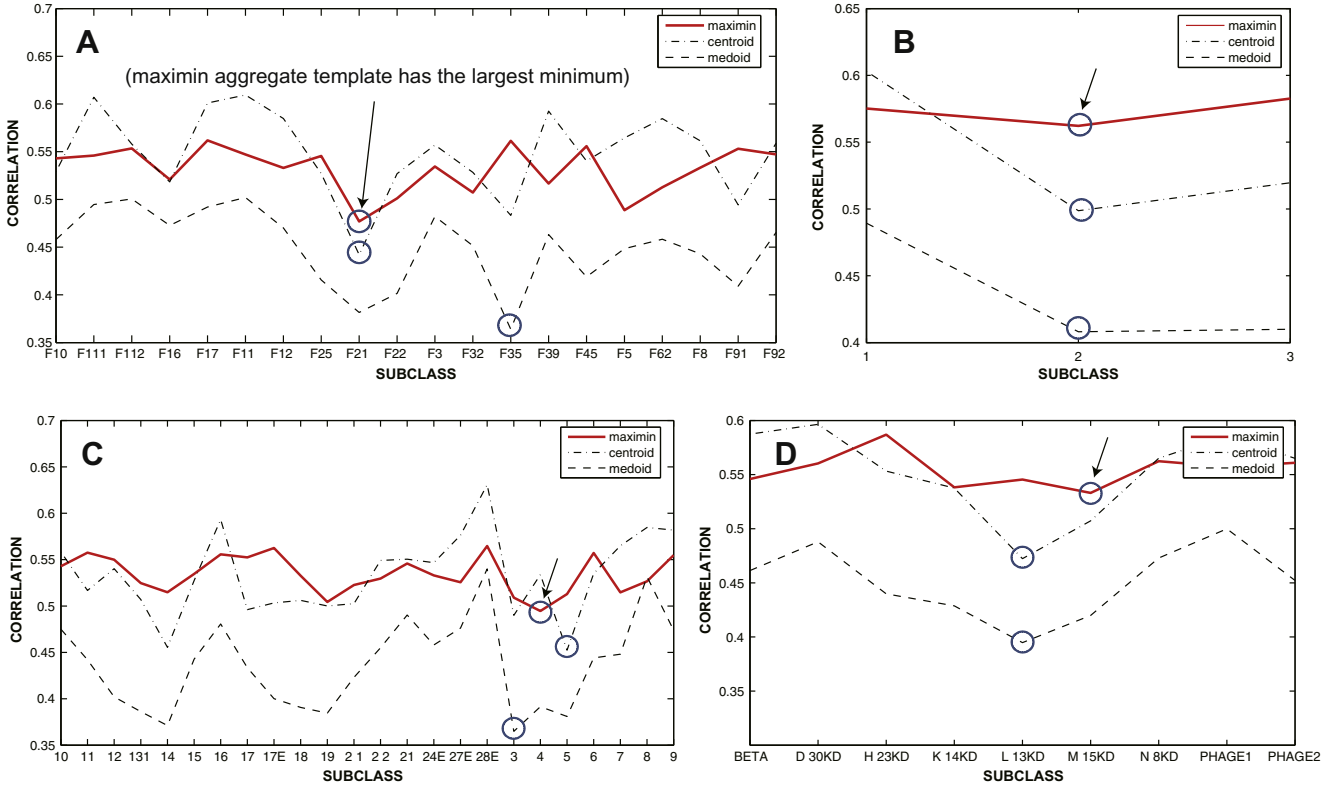

Fig. 2. The minimum correlation of the maximin aggregate template of a class with vectors of each subclass in the class:(A) GLYCOSYL_HYDROL, (B) TRANSFERRIN, (C) RIBOSOMAL_S and (D) RNA_POL. The minimum value of each curve is indicated by an open circle. Note that the circle on the curve representing the maximin template is located at the top in every plot. That is, the minimum correlation is maximized by using the maximin aggregate template.

clear visualization, the distribution of 3916 specificity and sensitivity values of each classifier was fitted to a bivariate Gaussian, and only its arithmetic mean and 0.7σ covariance ellipse were drawn in the plot.

As shown in Fig. 3A, the MCA-based classification outperformed the alternative methods by 7–19% and 6–27% in terms of average sensitivity and specificity, respectively. Notably, the MCA-based classifier showed better performance than the WebFEATURE method that is highly optimized to FEATURE vector classification. When compared with the other techniques than WebFEATURE, the proposed approach showed even greater performance advantages. The performance of SVM and the centroid- and medoid-based classifiers was similar. For SVM, the linear kernel [14] produced the best performance while the effect of other parameters such as the regularization cost [14] was negligible. For KNN, using $k = 8$ gave the best result. The effects of the parameters of SVM and KNN on classification performance can be found in Fig. 3B–D. In the experiment with 10 sampled data sets, the sensitivity and specificity of the proposed method was 24% and 11% better than the alternatives, respectively.

The next subsection presents some discussions on the relative performance of the aggregate-template-based classifiers.

3.3. Reasoning on the relative performance of template-based classifiers

In our experiments, using the maximin aggregate template produced better classification results than using the centroid or medoid aggregate template. We use Fig. 1B–D to explain possible reasons. In Fig. 1B, the directions of the templates for subclasses 1 and 2 of a class are indicated as black boxes with white subclass numbers inside. The maximin aggregate template of these subclasses is also shown, and it is identical to the centroid and medoid aggregate templates because there are only two subclass templates. However, the three aggregate templates become all distinct if another subclass exists in such a way that its template is located between those of subclasses 1and 2, as shown in Fig. 1C. In this case, misclassification of an unlabeled data can occur if the maximin aggregate template is not used, as depicted in Fig. 1D. Let θ_m, θ_c, and θ_o, respectively, denote the angles between the unlabeled vector and the following three aggregate templates: the maximin aggregate template, the centroid template, and the aggregate template (of any kind) of a different class. If $\theta_m < \theta_o < \theta_c$ as indicated in the figure, then using the centroid aggregate template (as well as the medoid aggregate template, whose angle with the unlabeled object is even greater than θ_c) would result in misclassification of the unlabeled vector, whereas using the maximin aggregate template would not.

3.4. A closer look at classification result

We looked for some unlabeled FEATURE vectors that are accurately classified by the maximin aggregate template, but misclassified by the other types of templates. We observed that some vectors in the class MOLYBDOPTERIN are misclassified into the classes HISTONE, HMG_COA_REDUCTASE, GLYCO_HORMONE or SIGMA_INTERACT by the centroid aggregate template, while they are correctly assigned by the maximin aggregate template. We also found a vector in the class INTERLEUKIN_1 that is misclassified into GLYCOSYL_HYDROL by the centroid and into THIOL_PROTEASE by the medoid, but accurately classified by the maximin-based method. For a more detailed example, Fig. 4A and B show a comparison of the microenvironments of two FEATURE vectors that have TRANSFERRIN (subclass 2) PROSITE annotation. The vectors are, in fact, originated from one protein (PDB ID 1h76), but from different alpha helical domains with 72.2% sequence identity. Although

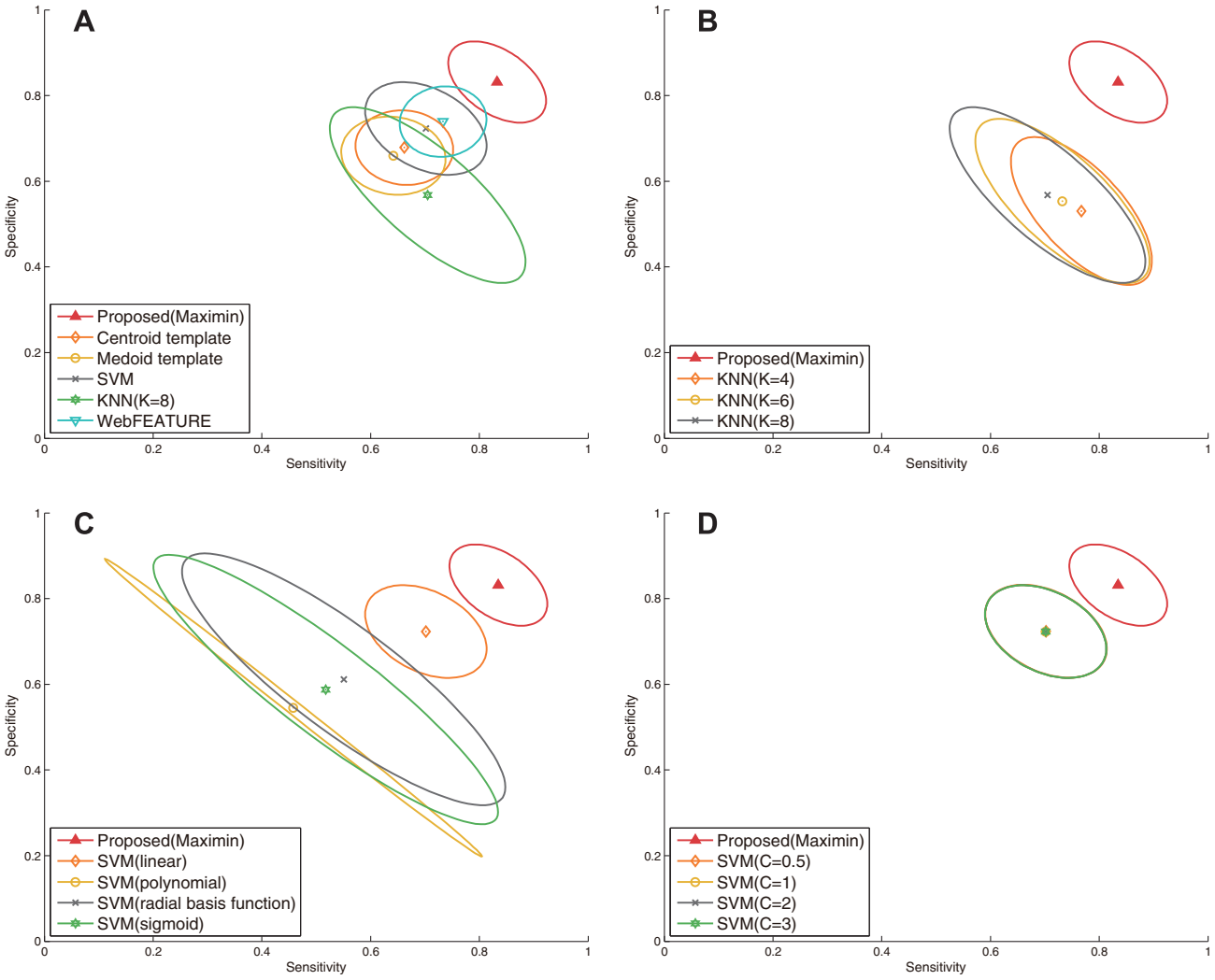

Fig. 3. Classification performance comparison in terms of sensitivity and specificity. Sensitivity (or true positive rate) is given by $\frac{TP}{TP+FN}$ where TP and FN represent the number of true positives and false negatives, respectively [16]. Specificity (or true negative rate) is defined as $\frac{TN}{FP+TN}$ where TN and FP mean the true negatives and false positives, respectively. For each type of classifier, the plot shows the 0.7σ covariance ellipse and mean of a bivariate Gaussian fitted to the distribution of 3916 sensitivity and specificity values. (A) Performance comparison. For SVM and KNN, the parameters producing the best result were used (*i.e.*, the linear kernel and the regularization cost of 1 for SVM and $k = 8$ for KNN). (B) The effect of k for KNN. (C) The effect of SVM kernels. (D) The effect of SVM regularization cost.

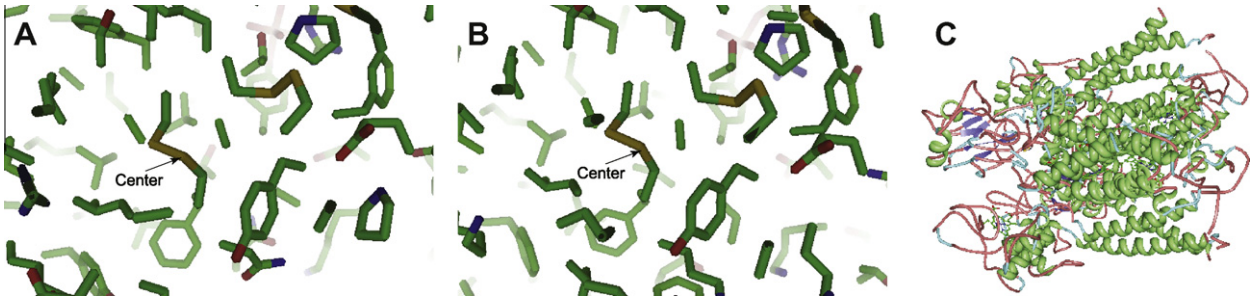

Fig. 4. Illustration of the microenvironment represented by a FEATURE vector that was correctly classified by using the maximin aggregate template but that was not by using the centroid aggregate template. (A) The microenvironment of the vector derived from the 10th helix of class TRANSFERRIN. The arrow indicates the residue around which the microenvironment was centered. Using either the maximin or centroid aggregate template resulted in the correct classification of this vector into TRANSFERRIN. (B) The microenvironment of the vector derived from the 26th helix of the same TRANSFERRIN class. Note that the two vectors depicted in (A) and (B) originate from the same protein but from different alpha helical domains therein with 72.2% sequence identity. Their microenvironments are nearly identical as seen above, and using the maximin aggregate template correctly classified both of them. In contrast, using the centroid aggregate template misclassified the vector in (B) into a similarly looking class, namely CYTOCHRMOME (PDB ID 1zrt), which has as many helices as TRANSFERRIN. (C) The structure of CYTOCHRMOME with alpha helices colored in green. The images shown were produced using Pymol [30] and the MBT Protein Workshop (http://mbt.sdsc.edu/software/applications).

these two vectors are derived from the same protein and share a great structural similarity, the vector shown in Fig. 4B is misclassified by the centroid aggregate template into the class CYTOCHROME that has many alpha helices (Fig. 4C). This result also supports our previous finding that the MCA-based classification technique has the highest correlation value among the three in case of TRANSFERRIN (subclass 2) in Fig. 2B. Even though the correlation values obtained by all three methods are the lowest for subclass 2, the maximin aggregate template assigns an uncharacterized vector into the right class while the centroid aggregate template does not.

4. Conclusions

Despite the desirable property of minimizing the misclassification risk in the correlation-based nearest neighbor classification, the maximin correlation analysis (MCA) has never been applied to the problem of automated protein function prediction. This article presents the first attempt to apply MCA to large-scale protein structure data represented in the FEATURE framework. Our experimental result indicates that MCA can significantly boost the classification performance of this FEATURE methodology. Given that correlation-based similarity measures have been widely used for mining protein data [27–29], we expect that MCA would be employed to enhance other types of automated function prediction frameworks.

Acknowledgment

This work was supported by the National Research Foundation (NRF) grants funded by the Korean Government Ministry of Engineering, Science and Technology (MEST) (Nos. 2010-0000631 and 2010-0000407).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.bbrc.2010.08.042.

References

[1] H.M. Berman, T. Battistuz, T.N. Bhat, W.F. Bluhm, P.E. Bourne, K. Burkhardt, Z. Feng, G.L. Gilliland, L. Iype, S. Jain, P. Fagan, J. Marvin, D. Padilla, V. Ravichandran, B. Schneider, N. Thanki, H. Weissig, J.D. Westbrook, C. Zardecki, The protein data bank, Acta Crystallogr., D Biol. Crystallogr. 58 (2002) 899–907.
[2] J. Westbrook, Z. Feng, S. Jain, T.N. Bhat, N. Thanki, V. Ravichandran, G.L. Gilliland, W. Bluhm, H. Weissig, D.S. Greer, P.E. Bourne, H.M. Berman, The protein data bank: unifying the archive, Nucleic Acids Res. 30 (2002) 245–248.
[3] H. Hegyi, M. Gerstein, The relationship between protein structure and function: a comprehensive survey with application to the yeast genome, J. Mol. Biol. 288 (1999) 147–164.
[4] Y. Loewenstein, D. Raimondo, O.C. Redfern, J. Watson, D. Frishman, M. Linial, C. Orengo, J. Thornton, A. Tramontano, Protein function annotation by homology-based inference, Genome Biol. 10 (2009) 207.
[5] P. Aloy, E. Querol, F.X. Aviles, M.J. Sternberg, Automated structure-based prediction of functional sites in proteins: applications to assessing the validity of inheriting protein function from homology in genome annotation and to protein docking, J. Mol. Biol. 311 (2001) 395–408.
[6] M.N. Wass, M.J. Sternberg, ConFunc – functional annotation in the twilight zone, Bioinformatics 24 (2008) 798–806.
[7] F. Pazos, M.J. Sternberg, Automated prediction of protein function and detection of functional sites from structure, Proc. Natl. Acad. Sci. USA 101 (2004) 14754–14759.
[8] O.C. Redfern, B.H. Dessailly, T.J. Dallman, I. Sillitoe, C.A. Orengo, FLORA: a novel method to predict protein function from structure in diverse superfamilies, PLoS Comput. Biol. 5 (2009) e1000485.
[9] M. Landau, I. Mayrose, Y. Rosenberg, F. Glaser, E. Martz, T. Pupko, N. Ben-Tal, ConSurf 2005: the projection of evolutionary conservation scores of residues on protein structures, Nucleic Acids Res. 33 (2005) W299–W302.
[10] R. Landgraf, I. Xenarios, D. Eisenberg, Three-dimensional cluster analysis identifies interfaces and functional residue clusters in proteins, J. Mol. Biol. 307 (2001) 1487–1502.
[11] T.R. Hvidsten, A. Laegreid, A. Kryshtafovych, G. Andersson, K. Fidelis, J. Komorowski, A comprehensive analysis of the structure-function relationship in proteins based on local structure similarity, PLoS One 4 (2009) e6266.
[12] N. Weinhold, O. Sander, F.S. Domingues, T. Lengauer, I. Sommer, Local function conservation in sequence and structure space, PLoS Comput. Biol. 4 (2008) e1000105.
[13] J.D. Watson, S. Sanderson, A. Ezersky, A. Savchenko, A. Edwards, C. Orengo, A. Joachimiak, R.A. Laskowski, J.M. Thornton, Towards fully automated structure-based function prediction in structural genomics: a case study, J. Mol. Biol. 367 (2007) 1511–1522.
[14] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, 2006.
[15] N. Cristianini, J. Shawe-Taylor, An Introduction To Support Vector Machines: And Other Kernel-based Learning Methods, Cambridge University Press, Cambridge, New York, 2000.
[16] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, second ed., Morgan Kaufman, Amsterdam, Boston, MA, 2005.
[17] H. Avi-Itzhak, J. Van Mieghem, L. Rub, Subclass pattern recognition: a maximin correlation approach, IEEE Trans. Pattern Anal. Mach. Intell. 17 (1995) 418–431.
[18] I. Halperin, D.S. Glazer, S. Wu, R.B. Altman, The FEATURE framework for protein function annotation: modeling new functions, improving performance, and extending to novel applications, BMC Genomics 9 (Suppl. 2) (2008) S2.
[19] M.P. Liang, D.R. Banatao, T.E. Klein, D.L. Brutlag, R.B. Altman, WebFEATURE: an interactive web tool for identifying and visualizing functional sites on macromolecular structures, Nucleic Acids Res. 31 (2003) 3324–3327.
[20] S. Yoon, J.C. Ebert, E.Y. Chung, G. De Micheli, R.B. Altman, Clustering protein environments for function prediction: finding PROSITE motifs in 3D, BMC Bioinformatics 8 (Suppl. 4) (2007) S10.
[21] B. Rosner, Fundmentals of Biostatistics, 7th ed., Cengage Learning, Boston, MA, 2010.
[22] S.P. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, UK, New York, 2004.
[23] N. Hulo, A. Bairoch, V. Bulliard, L. Cerutti, E. De Castro, P.S. Langendijk-Genevaux, M. Pagni, C.J. Sigrist, The PROSITE database, Nucleic Acids Res. 34 (2006) D227–D230.
[24] C.J. Sigrist, L. Cerutti, E. de Castro, P.S. Langendijk-Genevaux, V. Bulliard, A. Bairoch, N. Hulo, PROSITE, a protein domain database for functional characterization and annotation, Nucleic Acids Res. 38 (2010) D161–D166.
[25] C.J. Sigrist, L. Cerutti, N. Hulo, A. Gattiker, L. Falquet, M. Pagni, A. Bairoch, P. Bucher, PROSITE: a documented database using patterns and profiles as motif descriptors, Brief. Bioinform. 3 (2002) 265–274.
[26] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, 2001. Available from: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
[27] J.S. Bader, A. Chaudhuri, J.M. Rothberg, J. Chant, Gaining confidence in high-throughput protein interaction networks, Nat. Biotechnol. 22 (2004) 78–85.
[28] I.G. Choi, J. Kwon, S.H. Kim, Local feature frequency profile: a method to measure structural similarity in proteins, Proc. Natl. Acad. Sci. USA 101 (2004) 3797–3802.
[29] K. Lage, E.O. Karlberg, Z.M. Storling, P.I. Olason, A.G. Pedersen, O. Rigina, A.M. Hinsby, Z. Tumer, F. Pociot, N. Tommerup, Y. Moreau, S. Brunak, A human phenome-interactome network of protein complexes implicated in genetic disorders, Nat. Biotechnol. 25 (2007) 309–316.
[30] W.L. DeLano, The PyMol molecular graphics system, 2002. Available from: <http://www.pymol.org>.