BMC Bioinformatics

BioMed Central

Proceedings

Open Access

Choosing negative examples for the prediction of protein-protein interactions

Asa Ben-Hur*[1,2] and William Stafford Noble[3,4]

Address: [1]Department of Computer Science, Colorado State University, Fort Collins CO, USA, [2]Department of Computer Science, University of Colorado, Boulder CO, USA, [3]Department of Genome Sciences, University of Washington, Seattle WA, USA and [4]Department of Computer Science and Engineering, University of Washington, Seattle WA, USA

Email: Asa Ben-Hur* - asa@cs.colostate.edu; William Stafford Noble - noble@gs.washington.edu

* Corresponding author

from NIPS workshop on New Problems and Methods in Computational Biology
Whistler, Canada. 18 December 2004

Published: 20 March 2006

BMC Bioinformatics 2006, **7**(Suppl 1):S2 doi:10.1186/1471-2105-7-S1-S2

Abstract

The protein-protein interaction networks of even well-studied model organisms are sketchy at best, highlighting the continued need for computational methods to help direct experimentalists in the search for novel interactions. This need has prompted the development of a number of methods for predicting protein-protein interactions based on various sources of data and methodologies. The common method for choosing negative examples for training a predictor of protein-protein interactions is based on annotations of cellular localization, and the observation that pairs of proteins that have different localization patterns are unlikely to interact. While this method leads to high quality sets of non-interacting proteins, we find that this choice can lead to biased estimates of prediction accuracy, because the constraints placed on the distribution of the negative examples makes the task easier. The effects of this bias are demonstrated in the context of both sequence-based and non-sequence based features used for predicting protein-protein interactions.

Background

Despite advances in high-throughput experimental methods for detecting protein-protein interactions, the interaction networks for even well studied model organisms are far from complete. In addition, high throughput assays typically have a high rate of false positives [1]. Therefore, there is a continuing need for computational methods to complement existing experimental approaches.

Methods for predicting protein-protein interaction use a variety of data sources. Sequence-based methods are usually based on the domain, motif, or k-mer composition of the sequences. Sprinzak and Margalit [2] have noted that many pairs of structural domains tend to appear in interacting proteins, and have used this intuition to predict interactions according to the over-representation of pairs of domains. Domain and motif composition is also the basis of several Bayesian network models that aim to explain an observed interaction network in terms of interactions between pairs of motifs or domains [3-5]. In the context of kernel methods, similar kernels designed for predicting interactions from sequence were proposed in [6,7]. Other sequence-based methods use co-evolution of interacting proteins by comparing phylogenetic trees [8], correlated mutations [9], or gene fusion [10]. An alternative approach is to combine multiple sources of genomic information — gene expression, Gene Ontology annotations, transcriptional regulation, etc. — to predict co-membership in a complex [11-13].

All the above-mentioned methods require an informed choice of positive examples (interacting pairs of proteins)

Table 1: The dependence of ROC scores of several variables on the co-localization threshold for the MIPS/DIP interaction data. The variables are: GO process similarity, GO function similarity, and correlations between microarray data under various environmental conditions [19]. For each threshold we computed the average ROC scores for 10 drawings of the negative examples. The standard deviation is shown in parentheses.

threshold	GO process	GO function	microarray
1.00	0.81 (0.001)	0.64 (0.002)	0.64 (0.005)
0.50	0.82 (0.001)	0.65 (0.004)	0.64 (0.003)
0.20	0.82 (0.002)	0.66 (0.005)	0.65 (0.005)
0.10	0.83 (0.002)	0.66 (0.005)	0.66 (0.003)
0.04	0.83 (0.001)	0.67 (0.004)	0.66 (0.004)

and negative examples (non-interacting pairs of proteins) for training and assessing the performance of a classifier. In view of the large fraction of false positive interactions generated by high throughput methods, positive examples need to be chosen with care. These are often chosen as interactions generated by reliable methods (small scale experiments), interactions confirmed by several methods, or interactions confirmed by interacting paralogs [1,11,14,15].

Negative examples also need to be chosen with care, and two such selection methods have been described in the literature. Because there are no "gold standard" non-interactions, some authors suggest that high quality non-interactions can be generated by considering pairs of proteins whose cellular localization is different, most likely preventing the proteins from participating in a biologically relevant interaction [11,16]. Other authors use a simpler scheme, selecting non-interacting pairs uniformly at random from the set of all proteins pairs that are not known to interact [4,7,12,17].

In this paper, we argue that that the first method is not appropriate for assessing classifier accuracy. In particular, we show that restricting negative examples to non co-localized protein pairs leads to a biased estimate of the accuracy of a predictor of protein-protein interactions. The basic assumption underlying the assessment of the accuracy of a classifier is that the distribution of testing examples reflects the intended use of the method. In the case of predicting protein-protein interactions, a simple uniform random choice of non-interacting protein pairs yields an unbiased estimate of the true distribution. In contrast, imposing the constraint of non co-localization may induce a different distribution on the features that are used for classification. The resulting biased distribution of negative examples leads to over-optimistic estimates of classifier accuracy. This bias is likely to affect results reported in several papers [5,6,11].

The simpler selection scheme — choosing negative examples uniformly at random — also has potential pitfalls: because the interaction network is not complete, the set of negative examples can be contaminated with interacting proteins. This contamination, however, is likely to be very small: it has been estimated that the number of interactions in yeast is well below 100,000 [1,18], a number which is 0.25 percent of the total number of protein pairs in yeast. This effect is likely to be much smaller than the contamination of even high-quality positive examples; moreover, our results show that a support vector machine classifier is resistant to even higher levels of label contamination.

Results

In this paper we postulate that testing a classifier of protein-protein interactions on negative examples composed of pairs of proteins that are not co-localized results in a biased assessment of classifier accuracy. In order to test this hypothesis we need to define "co-localization." We do this using the subcellular localization component of the Gene Ontology (GO). GO keywords are becoming the standard in annotating gene products [20]. These keywords are arranged in a hierarchical manner in a rooted, directed acyclic graph, where keywords lower in the hierarchy represent more specific terms. Therefore, one cannot say that two proteins are not co-localized simply because they don't share the exact same GO terms. As a similarity measure between two GO terms we use the negative log of the fraction of genes annotated with the lowest common ancestor of the two terms. This similarity was introduced as a similarity measure on a hierarchy in [21], used in the context of GO annotations in [22], and used in a kernel in [7]. Using this measure of similarity allows us to generate parameterized sets of negative examples characterized by a maximum degree of similarity allowed between their GO cellular compartment annotations.

Perhaps the simplest way to predict protein-protein interactions is to represent pairs of proteins by a set of genomic features that reflect how likely they are to interact. Examples of features that were used for this task are similarity of GO process and GO function annotations, correlation of gene expression, presence of similar transcription factor binding sites in the upstream region of the genes, participation in common regulatory modules and so on [11-13].

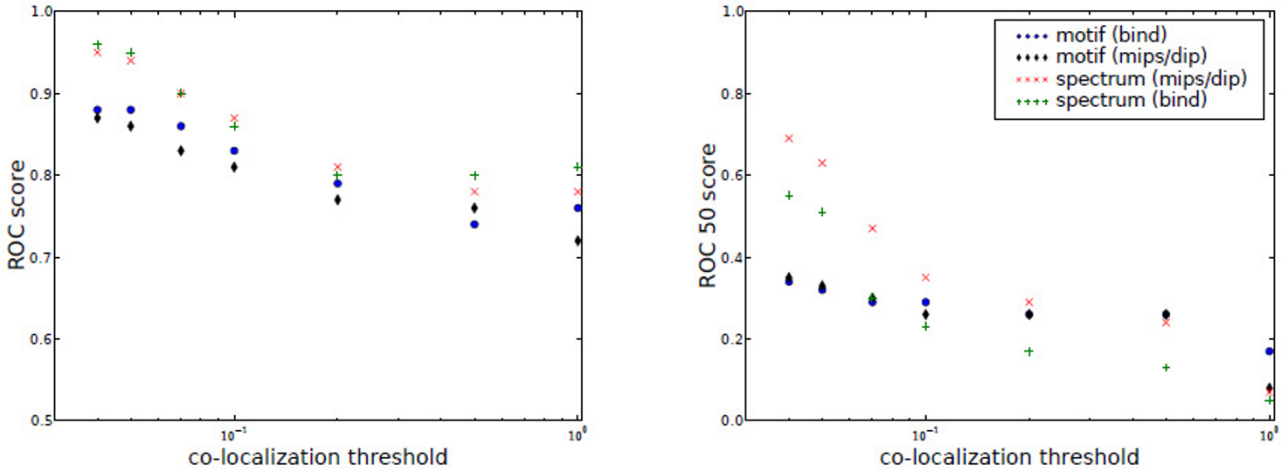

Figure 1
The dependence of prediction accuracy, quantified by the area under the ROC/ROC$_{50}$ curves, on the co-localization threshold used to choose negative examples. Enforcing the condition that no two proteins in the set of negative examples have a GO component similarity that is greater than a given threshold (the co-localization threshold) imposes a constraint on the distribution of negative examples. This constraint makes it easier for the classifier to distinguish between positive and negative examples, and the effect gets stronger as the co-localization threshold becomes smaller. All methods are SVM-based classifiers trained using different kernels on two interaction datasets. Results are computed using five-fold cross-validation, averaged over five drawings of negative examples. The spectrum kernel method uses pairs of k-mers as features; the motif method uses the composition of discrete sequence motifs, and the non-sequence method uses features such as co-expression as measured in microarray experiments, similarity in GO process and function annotations etc. We performed our experiment on two yeast physical interaction datasets: the BIND data is derived from the BIND database; the experiments using the non-sequence data were performed on a subset of reliable interactions that are found by multiple assays in BIND; DIP/MIPS is a dataset of reliable interactions derived from the DIP and MIPS databases.

Table 1 illustrates that as we vary the upper bound on the allowed similarity between the cellular compartment annotations of pairs of proteins in the negative examples (called the *co-localization threshold* in what follows), GO function and process annotations, and microarray data become more predictive of protein-protein interactions, as measured using the ROC score (the area under the receiver operating characteristic curve). This observation is not surprising. Consider, for example, biological process annotations. Interacting proteins often participate in similar processes. Conversely, negative examples that are not co-localized will be less likely to participate in similar biological processes, making this variable more predictive of interaction. A similar argument holds for the GO function annotations and gene expression correlations. Note that GO function annotations are less predictive than the process annotations because interactions are often required for carrying out a particular process, whereas proteins that carry out the same function can do so in different contexts, not requring interaction.

Using non-co-localized negative examples can lead to a bias when using sequence-based features as well. In this case the features are pairs of sequence features, e.g., motifs or k-mers that belong to a pair of protein sequences. Such a kernel was used in [6,7] with a support vector machine (SVM) classifier. The dimensionality of the feature space of these kernels is very high, and in fact, the method doesn't use an explicit representation of the features. For the sequence-based features we show the existence of the bias incurred by using non-co-localized negative examples by showing that the accuracy of a classifier depends on the co-localization threshold of the negative examples on which the method was tested. Figure 1 illustrates the increase in classifier accuracy as the co-localization threshold is decreased. This effect is much larger than the variability that results from the randomness in the choice of negative examples and the cross-validation (CV) estimate: the standard deviation of the ROC score on 10 drawings of the negative examples was 0.003, and the variability between different runs of CV is even lower. We can explain the higher accuracy for low co-localization threshold by the fact that the constraint on localization restricts the negative examples to a sub-space of sequence space, making the learning problem easier than when there is no constraint.

In our experiments we used sets of negative examples characterized by the similarity of the localization annotations of two proteins. To see the relevance of our results to other published work, we need to establish a relationship between our co-localization threshold, and criteria used elsewhere. The data of [11] is used in several studies of protein-protein interactions. They considered five very broad cellular compartments (cytoplasm, mitochondrion, nucleus, plasma membrane, and secretory pathway organelles). Four of these have corresponding nodes in the cellular compartment part of GO. The average GO similarity between these compartments ranges from 0.002 to 0.36, and is 0.13 on average. At this level of the co-localization threshold our results show a strong effect.

Discussion

There are many pitfalls in designing machine learning experiments (see [23] for an example in the context of feature selection). Design of experiments in the field of bioinformatics, where various sources of data are often correlated, requires special care to make sure no information on the testing example labels leaks to the representation of the training examples. In this paper, we illustrated a phenomenon where, by constraining the distribution of negative examples, the classification problem becomes easier. Although choosing negative examples as pairs of proteins that are localized to different cellular compartments creates high-quality negative examples, it also makes them easier to distinguish from interacting proteins. In the case where the data is characterized by features such as similarity of GO process or function annotations, constraining the distribution of the component similarity has a direct effect on the distribution of the GO process annotation.

In the case of the sequence-based classifiers, the improvement in classifier performance is the result of constraining the negative examples to a smaller region of sequence-space. We see a difference between the behavior of the motif/pfam kernels and the spectrum kernel: the results with the spectrum kernel are more strongly affected by the distribution of negative examples. We believe that this difference is the result of the greater flexibility of the spectrum kernel, which allows it to fit arbitrary training sets. The motif/pfam kernels, by contrast, use features that are more biologically relevant, so cannot be biased as much as the spectrum kernel. The gold standard negative examples of [11] were not only constrained by lack of co-localization; they also demanded that both pairs of proteins have GO annotations in both the function and process components. This constraint would likely increase classifier accuracy even further.

The reader may suspect that the improvement in classifier accuracy when constraining the negative examples to be non-co-localized may be the result of higher quality negative examples. To address this concern we performed the following experiment to test the effect of changing the labels of a small fraction of the negative examples. We considered the MIPS/DIP dataset with the spectrum kernel, and negative examples chosen with a co-localization threhsold of 0.1. We divided the dataset into two parts: training data (80%), test data (20%), and flipped the labels of 2% of the negative examples, a fraction likely to be much higher than the level of contamination under a choice of unconstrained selection of negative examples. SVMs were trained on both flipped and unflipped versions of the data. The average ROC (ROC_{50}) scores for 10 draws of the data were 0.874 (0.361) for the unflipped data, and 0.871 (0.356) for the flipped data. This experiment illustrates that SVMs can easily handle a larger amount of noise in the negative examples than is expected in the actual data. Thus, the effect shown above is not a result of better quality negative examples.

Without being aware of the bias in using gold standard non-interactions, one may think, looking at a couple of papers that describe methods for predicting protein protein interactions from sequence [5,6], that the problem is well addressed by these methods. However, this is not the case: the good performance is in fact a result of the biased selection of negative examples, and prediction of protein-protein interactions from sequence is a difficult problem that can still be considered unsolved.

Methods

Positive Examples

We focus on prediction of physical interactions in yeast and use interaction data derived from several sources. These interactions are used as positive examples when training our classifiers.

• Data from BIND [24]. BIND includes published interaction data from high-throughput experiments as well as curated entries derived from published papers. Using physical interactions yields a dataset comprised of 10,517 interactions among 4233 yeast proteins (downloaded July 9th, 2004). Selecting interactions that were verified by multiple experimental assays yields a dataset of 750 trusted interactions. We used all the interactions for training, but assessed the performance only on trusted interactions.

• A curated set of high quality interactions from MIPS and DIP [25,26], also used in [5]. This set contains MIPS interactions that were annotated as physical interactions derived from small scale experiments, DIP interactions from small scale experiments, and DIP interactions verified by multiple experiments, for a total of 4838 interactions.

In both cases we avoided using interactions that were validated by interacting paralogs in yeast to define trusted interactions, since those are likely to be easier to predict using the sequence-based methods. We eliminated self-interactions from each dataset, since many of the features we use are based on measures of similarity between the two proteins, e.g., gene expression correlation, and similarity of GO annotations.

Negative Examples

We compared two methods for choosing negative examples in this paper:

• Random pairs of proteins that are not known to physically interact.

• Parameterized sets of negative examples were chosen as random pairs of protein that are not known to physically interact, such that the similarity of their GO cellular compartment annotations is below some threshold.

In each case the number of negative examples was chosen to be equal to the number of positive examples in the dataset.

Support Vector Machines

The support vector machine (SVM) [27] is a classification method that provides state-of-the-art performance in many domains including bioinformatics [28,29]. SVMs access the data only through the *kernel function* which defines the similarity between data objects. This allows the use of SVMs even when an explicit vector-space representation of the data is not available, but a kernel function is provided. This is the case for one of the kernels used in this work, where a kernel between two pairs of sequences is defined (see below and [6,7]).

Figures of merit

In this paper we evaluate the accuracy of a trained classifier using two metrics. Both metrics — the area under the receiver operating characteristic curve (ROC score), and the normalized area under that curve up to the first 50 false positives, the ROC_{50} score — aim to measure both sensitivity and specificity by integrating over a curve that plots the true positive rate as a function of the false positive rate. The motivation for using both metrics is provided for example in [7].

Pairwise kernels

The kernels proposed in the literature for handling genomic information, e.g., sequence kernels such as the motif and spectrum kernels presented below, provide a similarity between two sequences, or more generally, a similarity between a representation of two proteins. Therefore, such kernels are not directly applicable to the task of predicting protein-protein interactions, which requires a similarity between two pairs of proteins. Thus, we want a function $K((X_1, X_2), (X_1', X_2'))$ that returns the similarity between the proteins X_1 and X_2 compared to the proteins X_1' and X_2'. We call a kernel that operates on individual genes or proteins a *genomic kernel*, and a kernel that compares pairs of genes or proteins a *pairwise kernel*. Two recent papers proposed an approach for converting a genomic kernel into a pairwise kernel [6,7]. They define the kernel

$$K((X_1, X_2), (X_1', X_2')) = K'(X_1, X_1')\,K'(X_2, X_2') + K'(X_1, X_2')\,K'(X_2, X_1'), \quad (1)$$

where $K'(\cdot, \cdot)$ is any genomic kernel. The intuition behind the kernel is that for the pair (X_1, X_2) to be considered similar to (X_1', X_2'), X_1 needs to be similar to X_1' and X_2 needs to be similar to X_2' (the first term) or X_1 is similar to X_2' and X_2 is similar to X_1' (the second term). The feature space for this kernel is a vector space of (symmetrized) pairs of features from the underlying genomic kernel.

Sequence kernels

We use two sequence kernels in this work: the spectrum kernel [30] and the motif kernel [31]. The spectrum kernel models a sequence in the space of all k-mers, and its feature space is a vector of counts of the number of times each k-mer appears in the sequence. For the motif kernel we use discrete sequence motifs, representing a sequence in terms of a motif composition vector that counts how many times a discrete sequence motif matches the sequence. To compute the motif kernel we used discrete sequence motifs constructed from the eBlocks database [32]. Yeast ORFs contain occurrences of 17,768 motifs out of a set of 42,718 motifs. For both kernels we used a normalized linear kernel in the space of k-mer/motif counts: $K(x,y)/\sqrt{K(x,x)K(y,y)}$.

Availability

Data and code related to this work are available at: http://noble.gs.washington.edu/proj/sppi. All the classification experiments were performed using the **PyML** machine learning library available at http://pyml.sourceforge.net.

Acknowledgements

This work is funded by NCRR NIH award P41 RR11823, by NHGRI NIH award R33 HG003070, and by NSF award BDI-0243257. WSN is an Alfred P. Sloan Research Fellow.

References

1. von Mering C, Krause R, Snel B, Cornell M, Olivier SG, Fields S, Bork P: **Comparative assessment of large-scale data sets of protein-protein interactions.** *Nature* 2002, **417:**399-403.
2. Sprinzak E, Margalit H: **Correlated sequence-signatures as markers of protein-protein interaction.** *Journal of Molecular Biology* 2001, **311:**681-692.
3. Deng M, Mehta S, Sun F, Chen T: **Inferring domain-domain interactions from protein-protein interactions.** *Genome Research* 2002, **12(10):**1540-1548.
4. Gomez SM, Noble WS, Rzhetsky A: **Learning to predict protein-protein interactions.** *Bioinformatics* 2003, **19:**1875-1881.
5. Wang H, Segal E, Ben-Hur A, Koller D, Brutlag DL: **Identifying Protein-Protein Interaction Sites on a Genome-Wide Scale.** In *Advances in Neural Information Processing Systems 17* Edited by: *Saul LK, Weiss Y, Bottou L. Cambridge, MA: MIT Press*; 2005:1465-1472.
6. Martin S, Roe D, Faulon JL: **Predicting protein-protein interactions using signature products.** *Bioinformatics* 2005, **21(2):**218-226.
7. Ben-Hur A, Noble WS: **Kernel methods for predicting protein-protein interactions.** *Bioinformatics* 2005, **21(suppl 1):**i38-i46.
8. Ramani A, Marcotte E: **Exploiting the co-evolution of interacting proteins to discover interaction specificity.** *Journal of Molecular Biology* 2003, **327:**273-284.
9. Pazos F, Valencia A: **In silico two-hybrid system for the selection of physically interacting protein pairs.** *Proteins: Structure, Function and Genetics* 2002, **47(2):**219-227.
10. Marcotte EM, Pellegrini M, Ng HL, Rice DW, Yeates TO, Eisenberg D: **Detecting protein function and protein-protein interactions from genome sequences.** *Science* 1999, **285:**751-753.
11. Jansen R, Yu H, Greenbaum D, Kluger Y, Krogan NJ, Chung S, Emili A, Snyder M, Greenblatt JF, Gerstein M: **A Bayesian networks approach for predicting protein-protein interactions from genomic data.** *Science* 2003, **302:**449-453.
12. Zhang LV, Wong S, King O, Roth F: **Predicting co-complexed protein pairs using genomic and proteomic data integration.** *BMC Bioinformatics* 2004, **5:**38-53.
13. Lin N, Wu B, Jansen R, Gerstein M, Zhao H: **Information assessment on predicting protein-protein interactions.** *BMC Bioinformatics* 2004, **5:**154.
14. Sprinzak E, Sattath S, Margalit H: **How Reliable are Experimental Protein-Protein Interaction Data?** *Journal of Molecular Biology* 2003, **327(5):**919-923.
15. Deane C, Salwinski L, Xenarios I, Eisenberg D: **Two Methods for Assessment of the Reliability of High Throughput Observations.** *Molecular & Cellular Proteomics* 2002, **1:**349-356.
16. Jansen R, Gerstein M: **Analyzing protein function on a genomic scale: the importance of gold-standard positives and negatives for network prediction.** *Current Opnion in Microbiology* 2004, **7:**535-545.
17. Qi Y, Klein-Seetharaman J, Bar-Joseph Z: **Random Forest Similarity for Protein-Protein Interaction Prediction from Multiple Sources.** *Proceedings of the Pacific Symposium on Biocomputing* 2005.
18. Grigoriev A: **On the number of protein-protein interactions in the yeast proteome.** *nar* 2003, **31(14):**4157-4161.
19. Gasch A, Spellman P, Kao C, Carmel-Harel O, Eisen M, Storz G, Botstein D, Brown P: **Genomic Expression Programs in the Response of Yeast Cells to Environmental Changes.** *Molecular Biology of the Cell* 2000, **11:**4241-4257.
20. Gene Ontology Consortium: **Gene ontology: tool for the unification of biology.** *Nat Genet* 2000, **25:**25-9.
21. Resnik P: **Using Information Content to Evaluate Semantic Similarity in a Taxonomy.** *IJCAI* 1995:448-453. [citeseer.ist.psu.edu/resnik95using.html]
22. Lord P, Stevens R, Brass A, Goble C: **Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation.** *Bioinformatics* 2003, **19(10):**1275-1283.
23. Ambroise C, McLachlan GJ: **Selection bias in gene extraction on the basis of microarray gene-expression data.** *Proceedings of the National Academy of Sciences of the United States of America* 2002, **99(10):**6562-6566.
24. Bader GD, Donaldson I, Wolting C, Ouellette BF, Pawson T, Hogue CW: **BIND-The Biomolecular Interaction Network Database.** *Nucleic Acids Res* 2001, **29:**242-245.
25. Mewes HW, Frishman D, Gruber C, Geier B, Haase D, Kaps A, Lemcke K, Mannhaupt G, Pfeiffer F, Schüller C, Stocker S, Weil B: **MIPS: a database for genomes and protein sequences.** *Nucleic Acids Research* 2000, **28:**37-40.
26. Xenarios I, Salwinski L, Duan XQJ, Higney P, Kim SM, Eisenberg D: **DIP: the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions.** *Nucleic Acids Research* 2002, **30:**303-305.
27. Boser BE, Guyon IM, Vapnik VN: **A Training Algorithm for Optimal Margin Classifiers.** *5th Annual ACM Workshop on COLT* 1992:144-152 [http://www.clopinet.com/isabelle/Papers/]. *Pittsburgh, PA: ACM Press*
28. Schölkopf B, Smola A: *Learning with Kernels Cambridge, MA: MIT Press*; 2002.
29. Noble WS: *Kernel methods in computational biology, chap. Support vector machine applications in computational biology Cambridge, MA: MIT Press*; 2004:71-92.
30. Leslie C, Eskin E, Noble WS: **The spectrum kernel: A string kernel for SVM protein classification.** In *Proceedings of the Pacific Symposium on Biocomputing* Edited by: *Altman RB, Dunker AK, Hunter L, Lauderdale K, Klein TE. New Jersey: World Scientific*; 2002:564-575.
31. Ben-hur A, Brutlag D: **Remote homology detection: a motif based approach.** *Proceedings of the Eleventh International Conference on Intelligent Systems for Molecular Biology* 2003, **19(suppl 1):**i26-i33.
32. Su Q, Liu L, Saxonov S, Brutlag D: **eBLOCKS: enumerating conserved protein blocks to achieve maximal sensitivity and specificity.** *Nucleic Acids Research* 2005, **33:**178-182.