Netw Model Anal Health Inform Bioinforma (2016) 5:26
DOI 10.1007/s13721-016-0134-5

ORIGINAL ARTICLE

Implementation and comparison of kernel-based learning methods to predict metabolic networks

Abiel Roche-Lima[1]

Received: 10 April 2016 / Revised: 29 June 2016 / Accepted: 1 July 2016 / Published online: 15 July 2016

Abstract Metabolic pathways can be conceptualized as the biological equivalent of a data pipeline. In living cells, series of chemical reactions are carried out by different proteins called enzymes in a stepwise manner. However, many pathways remain incompletely characterized, and in some of them, not all enzyme components have been identified. Kernel methods are useful in many difficult problem areas, such as document classification and bioinformatics. Specifically, kernel methods have been used recently to predict biological networks, such as protein–protein interaction networks and metabolic networks. In this paper, we implement and compare different methods and types of data to predict metabolic networks. The methods are Penalized Kernel Matrix Regression (PKMR) and pairwise Support Vector Machine (pSVM). We develop several experiments using these methods with sequence, non-sequence, and combined data. We obtain better accuracy when the sequence data are used in both methods. Whereas when the methods are compared using the same type of data, the pSVM approach shows better accuracy. The best results are obtained with pSVM using all combined kernels.

Keywords Network prediction · Metabolic pathways · Machine learning · Kernel methods

✉ Abiel Roche-Lima
abiel.roche@upr.edu

[1] Collaboration Center for Research in Health Disparities, Medical Science Campus, University of Puerto Rico., PO Box 365067, San Juan, PR 00936-5067, USA

1 Introduction

Biochemical pathways are chemical reactions in the cell where enzymes catalyse reactions to produce other compounds based on substrates. For example, in the metabolic pathway that involves glycolysis, the glucose is broken down into smaller products, such as carbon dioxide and water (Luo et al. 2007). Finding the enzymes involved in the reactions and their interactions is still a very challenging topic. The development of pathway databases, such as KEGG (Kanehisa et al. 2008) and EcoCyc (Latendresse et al. 2012), has increased the current knowledge about metabolic networks. Using these databases, methods based on gene annotations are used to predict metabolic networks (Latendresse et al. 2012; Karp et al. 2011). However, current genome annotation pipelines may fail to assign identities correctly to score genes and to detect other genes altogether. Thus, metabolic network prediction algorithms using current genome annotation pipelines may predict inaccurate interactions, for example, the Pathway Tools described by Karp et al. (2011).

To infer metabolic networks, supervised learning approaches have been developed in the framework of kernel methods by Kotera et al. (2013), such as Support Vector Machines (SVMs). While SVMs are a classical paradigm in machine learning, they cannot be directly applied to the biological network inference problems, since the goal is to predict pair of genes (Ben-Hur and Noble 2005). Thus, the pairwise Support Vector Machine (pSVM) approach is used instead (Oyama and Manning 2004). Vert et al. (2007) and Kashima et al. (2010) apply pSVM methods to predict metabolic networks, but only combine non-sequence data. In addition, Roche-Lima et al. (2014) use sequence kernels (i.e., PRK—Pairwise Rational Kernels) combined with SVM methods and obtain good

accuracy values and execution times, but do not compare with non-sequence kernels.

There are other supervised learning algorithms, such as Kernel Canonical Correlation Analysis (KCCA) (Yamanishi et al. 2004) and Penalized Kernel Matrix Regression (PKMR) (Yamanishi and Vert 2007), which are computationally more efficient, but they lack the ability to give precise predictions. In addition, these algorithms have only been reported in the literature using non-sequence kernels (Yamanishi 2010; Kotera et al. 2012).

In our research, we consider these problems, implementing methods to predict metabolic networks based on raw data directly related to the sequence information (e.g., nucleotides and protein sequences). We hypothesize that sequence kernels, created from raw sequence data, will be more precise that non-sequence kernels to predict metabolic networks. We then implement two of the supervised learning methods (i.e., PKMR and pSVM), and for first time, we compare these two methods combined with sequence and non-sequence kernels.

2 Materials and methods

2.1 Metabolic networks

Metabolic networks were represented as a graph, where vertices (nodes) were the enzymes, and the edges (branches) were the enzyme–enzyme relations (proteins catalyzing two continuous reactions in a pathway). Traditionally, metabolic pathway representations considered enzymes as vertices, and metabolites as edges. To avoid confusion, our graphs represented interactions between pairs of enzymes as discrete data points similar to Yamanishi (2010). An example of the graph representation can be seen in (Roche-Lima et al. 2014, Fig. 2).

2.2 Data

We used information of the yeast *Saccharomyces cerevisiae* (Sikorski and Hieter 1989) taken from the KEGG pathway databases (Kanehisa et al. 2008). This species was selected, because it was a well-studied organism with several defined models to predict biological networks. Moreover, other kernel methods had been described and tested using data from this species (Ben-Hur and Noble 2005; Kashima et al. 2010; Yamanishi 2010). As a training set, we used 5149 interactions from 755 known genes. A graph was built based on these interactions (training set) as a representation of the metabolic networks of the yeast *Saccharomyces cerevisiae*. Then, this graph and other related data sets were converted into kernels.

Kernels allowed working in a unified mathematical framework across different types of data. A kernel was a measure of similarity that satisfied the additional condition of being a dot product in some feature space (see Scholkopf and Smola 2002 for details). Data were represented as a positive definite kernel K that was a symmetric function $K : X^2 \rightarrow \mathbb{R}$ that satisfied $\sum_{i,j}^{n} a_i a_j K(X x_i X_j)$ and $(a_1, a_2, \ldots, a_n) \in \mathbb{R}^n$, where X was the set of entities.

2.2.1 Non-sequence data

In our context, non-sequence kernels manipulate data that were binary or numerical. We used three different types of non-sequence data, i.e., gene expression, gene localization, and phylogenetic data. All these data have been used in other research as kernels (Vert et al. 2007; Kashima et al. 2010; Yamanishi 2010). Gene expression data were obtained by Yamanishi (2010) using the results from 157 microarray experiments (Spellman et al. 1998; Eisen et al. 1998). Each gene was associated with a 157-element numerical vector that represented the results from the experiments. Gaussian Radial Bases Function (RBF) Kernel was used to manipulate this data, and we defined the same parameters that Yamanishi (2010) used in their experiments. We denoted the final kernel as k_{exp}.

The gene localization data were represented as a 23-element binary vector for each gene, following Yamanishi (2010). A total of 23 intracellular localizations were defined (e.g., mitochondrion, Golgi, nucleus, and others). The value was 1, if the gene was present in the intracellular localization or 0 otherwise. Similar to Yamanishi (2010), we used the linear kernel applied to these data with the same parameters. We denoted this kernel as k_{loc}.

The phylogenetic profile data were obtained from 145 organisms, which describe the set of orthologous genes. These organisms were selected based on the criteria defined in Yamanishi (2010). Each gene was associated with a 145-element binary vector. The value was 1, if the gene was present in this organism or 0 otherwise. A Gaussian RBF kernel was used to compute this data with the same parameters used by Yamanishi (2010). This final kernel was denoted as k_{phy}.

2.2.2 Sequence data

Sequence kernels defined similarities over finite sequences of symbols with different lengths. The sequence data were then converted to sequence kernels. In our research, we used three sequence kernels, Pfam, Motif, and Spectrum, defined by Ben-Hur and Noble (2005). We chose these

sequence kernels to be able to compare our results with the previous published works, such as Yu et al. (2010); Ben-Hur and Noble (2005); Roche-Lima et al. (2014), and Allauzen et al. (2008).

The Pfam kernel (Gomez et al. 2003) was computed based on a set of Hidden Markov Models (HMMs), where each gene that codes for an enzyme was compared with every HMM in the Pfam database. The E value statistics were obtained as features for the 13,672 domain HMMs in the Pfam version 26.0 (Punta et al. 2012). Thus, each protein was represented by a vector of 13,672 log E values, and the kernel was computed based on these vectors (see Allauzen et al. (2008) for more details). We denoted this kernel as k_{pfam}.

The Motif kernel (Ben-Hur and Brutlag 2003) was also used. It was obtained by calculating how many times a discrete sequence motif matched each of the protein sequences. The eMotif database (Huang and Brutlag 2001) was used to extract the discrete sequence motifs. A vector of E values was associated for each of the proteins (genes coding for the proteins). The kernel was finally computed as dot products of those vectors (see Ben-Hur and Noble 2005 for more details). The kernel was called k_{motif}.

Finally, the Spectrum kernel defined by Leslie et al. (2004) was also considered. This kernel represented sequence similarities by counting how many times an n-gram (k_{mer}) appeared in each of the pairs of sequences. Each gene had an associated featured vector of n-gram counts (we considered $n = 3$). Similar to the data above, the kernel was computed to represent the dot products using the associated feature vectors. We denoted this kernel as k_{ngram}.

2.2.3 Combined data

We also computed the linear combination of the kernels described above, representing the heterogeneous data combination. We used different types of data to predict metabolic networks. $K_1, \ldots, K_n$ were the kernels that represented the data, so K_n corresponded to the n-th data set. Yamanishi (2010) mentioned the advantages of considering the linear combination as weighted sum of kernels, i.e., $\sum_{n=1}^{N} W_n K_n$, where W_n was a weight (real coefficient) associated to the kernel K_n. The coefficients should be related to the importance of the data set n for the prediction method. In our research, we considered the weights (W_n) as the accuracy values obtained during the inference process using the individual kernel K_n, i.e., ROC score Yamanishi et al. (2005). In future studies, weight values may be computed in different ways.

2.3 Methods

We used kernel-based supervised learning network inference methods to predict biological networks based on kernel frameworks. First, part of the network (with known interactions—training set) was used during the learning inference process to obtain the model. Second, new interactions were predicted using the model. In machine learning, supervised classifications are a classical paradigm. However, it could not be applied directly to the problem of network inference, because our goal was to predict relations between pairs of nodes, not individual nodes (Yamanishi 2010). Therefore, we first define the pairwise kernel and, later, the methods PKMR and pSVM.

2.3.1 Pairwise kernels

The kernels described in the sections above provide similarities between simple enzymes. In our experiments, we used a different type of kernel called pairwise kernel (Pahikkala et al. 2012; Brunner et al. 2012) that provide similarity measures for pairs of entities. The general pairwise kernel was represented as $K : (X \times X) (X \times X) \rightarrow \boldsymbol{R}$, where X is a set of vertices (enzymes) and $\boldsymbol{R}$ is a set of real values. In this research, we used the Pairwise Tensor Product Kernel or Kronecker Kernel (Basilico and Hofmann 2004; Oyama and Manning 2004; Ben-Hur and Noble 2005) that is computed as $K\big((X_1, X_2), (X_1', X_2')\big) = k'(X_1, X_1')k'(X_2, X_2') + k'(X_1, X_2') k'(X_2, X_1')$, where k' is a simple kernel and X_1, X_2, X_1', X_2' are the enzymes (k' represent any of the kernel described in the previous sections).

2.3.2 Penalized kernel matrix regression (PKMR)

Kernel Matrix Regression methods were based on the supervised graph inference framework to predict metabolic networks with metric learning. A formalism of the problem can be defined as follows:

> given an undirected graph $\Gamma = (V, E)$, with a set of vertices $V = (V_1, V_2, \ldots, V_n)$ and a set of edges $E \subset (V \times V)$,
> then, for an additional set of vertices $V' = V_1', V_2', \ldots, V_n'$, the goal was to infer the set of new edges $E' \subset V' \times (V + V') \cup (V + V') \times V'$, that involved the additional vertices in V'.

Yamanishi et al. (2005) described methods to solve this problem, such as KCCA, PKMR, Kernel Matrix Completion, and Expectation-Maximization algorithms. He obtained as a result that the method with the best accuracy

was PKMR (Yamanishi and Vert 2007), a modified version of Kernel Matrix Regression method.

In our research, we implemented PKMR using the R library (R Core Team 2013). To make the data compatible with the pSVM method, the chemical compatibility network was not taken into consideration (Yamanishi 2010), since we aimed to compare data directly related to the genes. Future implementations may include this information.

2.3.3 *Pairwise support vector machine (pSVM)*

pSVM methods classified whether a pair (x_1, y_1) belonged to the same category or to a different one. Then, while SVM methods classified simple entities, pSVM methods classified pairs of entities. pSVM was defined by Brunner et al. (2012) as follows:

> given a training data set $((x_i, y_i), d_i)$, d_i with binary classification values (i.e., (x_i, y_i) classified as $+1$ or (x_i, y_i) classified as -1), $i = 1, \ldots, n$ and the function Φ,
> then, a pSVM method found an optimal hyperplane, i.e., $w^T \Phi(x_i, y_i) + b = 0$, where the points were separated into two categories.

We implemented programs to apply pSVM to predict the metabolic networks with our data sets, using LIBSVM (Chang and Lin 2011) and Python Machine Learning (PyML) (Ben-Hur et al. 2008) libraries.

2.4 Experiments

We developed six groups of experiments using different data and methods (see Table 1 for more details).

For evaluation, we used the area under the ROC curve (AUC score) (Gribskov and Robinson 1996) to measure the accuracy. It was defined as a function of the rates of true-positives (predicted enzymes pairs were present in the data set) and false-positives (predicted protein pairs were absent in the data set). A stratified cross-validation procedure was used with fold equal to 10 (tenfold cross-validation) (Kohavi et al. 1995). We also collected execution times (Time s). All experiments were run using a computer with a microprocessor Intel i7CORE and RAM memory of 8 MB.

In addition, we computed the 95 % confidence intervals (CIs) for average AUC scores. We used a distribution-independent technique proposed by Cortes and Mohri (2005). As they described, the variance depends on the number of positive and negative examples in the training set and the number of errors during the classification process. In our case, the training set consisted in 2575 positive and 2574 negative interactions, out of 5149 total interactions. The errors in the classification process ranged between 750 and 1851.

3 Results and discussion

3.1 Comparing data

When we compare sequence and non-sequence data, within the same supervised learning method, better accuracy values are obtained with the sequence kernel (see Table 2, Experiments II–PKMR–Sequence and V–pSVM–Sequence). For example, in the PKMR method, the accuracy value is improved from AUC = 0.503 (the lowest value in experiment I–PKMR–Non-Sequence kernel) to AUC = 0.821 (the highest value in experiment II–PKMR–Sequence kernel). This proves our hypothesis about better accuracy values when sequence kernels are used, since errors from the genome annotation process are bypassed. However, the execution times for the methods using sequence kernels are more than doubled when they are compared with non-sequence kernels (i.e., Time = 240 s—the lowest time in experiment I–PKMR–Non-Sequence versus Time = 530 s—the highest time in experiment II–PKMR–Sequence). This is because computing sequence kernels (i.e., k_{pfam}, k_{motif}, and k_{mer}) consume more computational resources.

The best results are obtained with the kernels that represent the combined heterogeneous data within the same supervised learning method, i.e., Table 2, experiment III–PKMR–Combined and VI–pSVM–Combined. In this case, for the PKMR method, the accuracy is improved from AUC = 0.797 (i.e., the best accuracy using the simple kernel—k_{pfam} in Experiment II) to AUC = 0.840 (i.e., weighted kernel: $w_1 k_{\text{exp}} + w_2 k_{\text{loc}} + w_3 k_{\text{phy}} + w_4 k_{\text{pfam}} + w_5 k_{\text{motif}} + w_3 k_{\text{mer}}$ in

Table 1 Experiments are grouped by methods (experiment I, II, III—PKMR and experiment IV, V, VI—pSVM) and by type of data (I, IV—non-sequence data, II, V—sequence data, and III, VI—combined data)

Experiment	Methods	Type of kernel
I	PKMR	Non-sequence (described in Sect. 2.2.1)
II	PKMR	Sequence (described in Sect. 2.2.2)
III	PKMR	Combined sequence and non-sequence (described in Sect. 2.2.3)
IV	pSVM	Non-sequence (described in Sect. 2.2.1)
V	pSVM	Sequence (described in Sect. 2.2.2)
VI	pSVM	Combined sequence and non-sequence (described in Sect. 2.2.3)

Table 2 Results collected during the experiments

Experiment	Predictor kernel	AUC score	Time s	Confidence intervals
I–PKMR–Non-Sequence	k_{exp}	0.660	300	[0.655, 0.665]
	k_{loc}	0.503	240	[0.499, 0.507]
	k_{phy}	0.775	240	[0.771, 0.779]
	$k_{exp} + k_{loc} + k_{phy}$	0.755	350	[0.752, 0.759]
	$w_1k_{exp} + w_2k_{loc} + w_3k_{phy}$	0.799	420	[0.791, 0.807]
II–PKMR–Sequence	k_{pfam}	0.797	450	[0.793, 0.801]
	k_{motif}	0.782	430	[0.778, 0.786]
	k_{mer}	0.725	420	[0.720, 0.731]
	$k_{pfam} + k_{motif} + k_{mer}$	0.817	480	[0.811, 0.823]
	$w_4k_{pfam} + w_5k_{motif} + w_6k_{mer}$	0.821	530	[0.818, 0.824]
III–PKMR–Combined	$k_{phy} + k_{pfam}$	0.812	470	[0.809, 0.816]
(sequence and non-sequence)	$k_{exp} + k_{loc} + k_{phy} + k_{pfam} + k_{motif} + k_{mer}$	0.831	610	[0.828, 0.834]
	$w_1k_{exp} + w_2k_{loc} + w_3k_{phy} + w_4k_{pfam} + w_5k_{motif} + w_6k_{mer}$	0.840	720	[0.831, 0.849]
IV–pSVM–Non-Sequence	k_{exp}	0.791	9020	[0.786, 0.796]
	k_{loc}	0.696	7800	[0.692, 0.700]
	k_{phy}	0.802	7980	[0.797, 0.807]
	$k_{exp} + k_{loc} + k_{phy}$	0.818	10,100	[0.812, 0.824]
	$w_1k_{exp} + w_2k_{loc} + w_3k_{phy}$	0.877	10,121	[0.871, 0.883]
V–pSVM–Sequence	k_{pfam}	0.887	12,060	[0.879, 0.895]
	k_{motif}	0.868	12,000	[0.859, 0.877]
	k_{mer}	0.840	11,760	[0.836, 0.844]
	$k_{pfam} + k_{motif} + k_{mer}$	0.898	12,220	[0.891, 0.905]
	$w_4k_{pfam} + w_5k_{motif} + w_6k_{mer}$	0.910	12,800	[0.901, 0.919]
VI–pSVM–Combined	$k_{phy} + k_{pfam}$	0.890	12,100	[0.882, 0.898]
(Sequence and non-sequence)	$k_{exp} + k_{loc} + k_{phy} + k_{pfam} + k_{motif} + k_{mer}$	0.939	13,420	[0.935, 0.944]
	$w_1k_{exp} + w_2k_{loc} + w_3k_{phy} + w_4k_{pfam} + w_5k_{motif} + w_6k_{mer}$	0.940	14,010	[0.934, 0.946]

These are AUC score (area under the ROC curve as accuracy), time s (Execution times in seconds), and confidence intervals

Experiment III). Likewise, when we use only the Pfam—k_{pfam} kernel (AUC = 0.797 in experiment II), the best accuracy is obtained. Then, we also test this simple sequence kernel combined with other simple kernels. The best results are obtained combining the Pfam—k_{pfam} and phylogenetic—k_{phy} kernels (see the accuracy values of $k_{phy} + k_{pfam}$ kernel in Table 2, experiment III–PKMR–Combined and VI-pSVM–Combined). This result coincides with Allauzen et al. (2008), where they stated "the importance of the phyletic retention feature as a possible reason for the superior performance of the combined kernel compared with Pfam alone".

3.2 Comparing methods

As can be seen in Table 2, pSVM methods (experiment IV–pSVM–Non-Sequence, V–pSVM–Sequence, and VI–pSVM–Combined) outperform the precision values of PKMR method (experiment I–PKMR–Non-Sequence, II–PKMR–Sequence and III–PKMR–Combined). For example, using the PKMR method, the AUC score for k_{exp} (Experiment I–PKMR–Non-Sequence) is 0.660 compared to 0.791 (experiment III–PKMR–Combined). However, the execution times are considerably increased for pSVMs (see Table 2 Times values for experiments I–PKMR–Non-Sequence and II–PKMR–Sequence in comparison with experiments IV–pSVM–Non-Sequence and V–pSVM–Sequence). Processing pSVM involves more computational resources than PKMR methods; however, better accuracy values are obtained. In all the cases, the confidence intervals are above the behaviour of a random classifier.

Figure 1 represents the results for both methods (PKMR and pSVM) using only sequence kernels. Although the most time consuming method is pSVM, it provides an important improvement in the accuracy values [the peaks are reached combining the sequence kernel ($k_{pfam} + k_{motif} + k_{ngram}$) and pSVM method]. Yamanishi (2010) mentions these expected high processing times for SVM methods, but never tested them to evaluate how the accuracy could be improved. Roche-Lima et al. (2014) use a

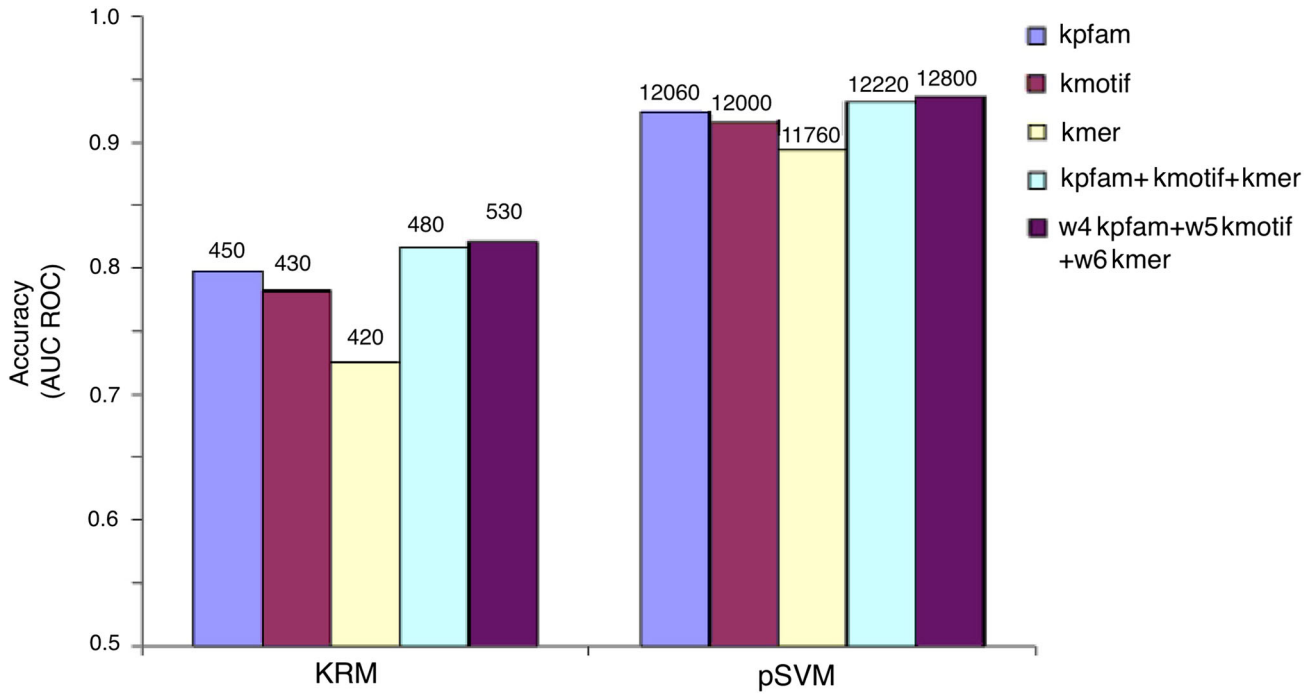

Fig. 1 Comparison of the methods (PKMR—Penalized Kernel Matrix Regression and pSVM—pairwise Support Vector Machine) for the sequence data kernels, related to accuracy and execution times

different representation of the sequence kernel and decrease execution time of the sequence kernel computation; however, they still use existing SVM methods. In addition, the accuracy values obtained in this research are better than the values reported by Roche-Lima et al. (2014). Thus, we consider that pSVM implementation can be optimized to obtain better processing time and to maintain these good accuracy values. Likewise, sequence kernel representations can be also optimized to combine with pSVM methods to improve both performance and accuracy.

4 Conclusion

We developed, for the first time, experiments using sequence data with PKMR and pSVM methods to predict metabolic networks. We proved that the best accuracy values were obtained using sequence kernels. This was because other tools to predict metabolic networks were based on the gene annotations (Latendresse et al. 2012; Karp et al. 2011). As we used raw sequence data (represented as sequence kernels), it bypassed the annotations and the errors associated with these steps.

We also proved that pSVM methods were more precise than PKMR methods. The best accuracy values were obtained when pSVM methods were combined with sequence kernels. However, pSVM methods were very expensive in terms of computational resources, such as execution times. pSVM methods required even more computational resources when using sequence kernels.

In future works, pSVM method can be optimized using other implementations, such as Dual Coordinate Descent algorithm combined with rational kernels to manipulate sequence data (Allauzen et al. 2011). As well, a parallel implementation could be used to improve performance (Tyree et al. 2014).

Acknowledgments I would like to acknowledge Dr. Michael Domaratzki and Dr. Brian Fristensky from the University of Manitoba, Canada, for their guidance and suggestions during the development of this research. This work was partially funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) at the University of Manitoba, Canada), and RCMI Grant G12 MD007600 (National Institute on Minority Health and Health Disparities) from the National Institutes of Health, at the University of Puerto Rico.

Compliance with ethical standards

Conflict of interest The author declares that there is no conflict of interest regarding the publication of this manuscript.

References

Allauzen C, Mohri M, Talwalkar A (2008) Sequence kernels for predicting protein essentiality. In: Proceedings of the 25th international conference on machine learning ICML '08. 9–16. ACM New York, NY, USA

Allauzen C, Cortes C, Mohri M (2011) A dual coordinate descent algorithm for SVMs combined with rational kernels. Int J Found Comput Sci 22:1761–1779

Basilico J, Hofmann T (2004) Unifying collaborative and content based filtering. In: Brodley C (ed) Proceedings of the twenty-first international conference on machine learning ACM p 9

Ben-Hur A, Brutlag D (2003) Remote homology detection: a motif based approach. Bioinformatics 19:i26–i33

Ben-Hur A, Noble WS (2005) Kernel methods for predicting protein–protein interactions. Bioinformatics 21:i38–i46

Ben-Hur A, Ong CS, Sonnenburg S, Scholkopf B, Ratsch G (2008) Support vector machines and kernels for computational biology. PLoS Comput Biol 4:e1000173

Brunner C, Fischer A, Luig K, Thies T (2012) Pairwise support vector machines and their application to large scale problems. J Mach Learn Res 13:2279–2292

Chang CC, Lin CJ (2011) LibSVM: a library for support vector machines. ACM Trans Intell Syst Technol (TIST) 2:27

Cortes C, Mohri M (2005) Confidence intervals for the area under the ROC curve. Advances in neural information processing systems. Curran Associates, UK, p 305

Eisen MB, Spellman PT, Brown PO, Botstein D (1998) Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci 95:14863–14868

Gomez SM, Noble WS, Rzhetsky A (2003) Learning to predict protein–protein interactions from protein sequences. Bioinformatics 19:1875–1881

Gribskov M, Robinson NL (1996) Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. Comput Chem 20:25–33

Huang JY, Brutlag DL (2001) The emotif database. Nucleic Acids Res 29:202–204

Kanehisa M, Araki M, Goto S, Hattori M, Hirakawa M, Itoh M, Katayama T, Kawashima S, Okuda S, Tokimatsu T et al (2008) KEGG for linking genomes to life and the environment. Nucleic Acids Res 36:D480–D484

Karp PD, Latendresse M, Caspi R (2011) The pathway tools pathway prediction algorithm. Stand Genom Sci 5:424–429

Kashima H, Oyama S, Yamanishi Y, Tsuda K (2010) Cartesian kernel: an efficient alternative to the pairwise kernel. IEICE Trans Inf Syst 93:2672–2679

Kohavi R et al (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. IJCAI 14:1137–1145

Kotera M, Yamanishi Y, Moriya Y, Kanehisa M, Goto S (2012) GENIES: gene network inference engine based on supervised analysis. Nucleic Acids Res 40(W1):162–167

Kotera M, Tabei Y, Yamanishi Y, Tokimatsu T, Goto S (2013) Supervised reconstruction of metabolic pathways from metabolome-scale compound sets. Bioinformatics 29:i135–i144

Latendresse M, Paley S, Karp PD (2012) Browsing metabolic and regulatory networks with BioCyc. In: van Helden J (ed) Bacterial molecular networks. Springer, New York, pp 197–216

Leslie CS, Eskin E, Cohen A, Weston J, Noble WS (2004) Mismatch string kernels for discriminative protein classification. Bioinformatics 20:467–476

Luo B, Groenke K, Takors R, Wandrey C, Oldiges M (2007) Simultaneous determination of multiple intracellular metabolites in glycolysis, pentose phosphate pathway and tricarboxylic acid cycle by liquid chromatography–mass spectrometry. J Chromatogr A 1147(2):153–164

Oyama S, Manning CD (2004) Using feature conjunctions across examples for learning pairwise classifiers. In: Boulicaut J-F (eds) European conference on machine learning. Springer, Berlin, Heidelberg, pp 322–333

Pahikkala T, Airola A, Stock M, De Baets B, Waegeman W (2012) Efficient regularized least-squares algorithms for conditional ranking on relational data. Machine Learning 93:321–356

Punta M, Coggill PC, Eberhardt RY, Mistry J, Tate J, Boursnell C, Pang N, Forslund K, Ceric G, Clements J et al (2012) The Pfam protein families database. Nucleic Acids Res 40:D290–D301

R Core Team (2013) R: a language and Environment for Statistical Computing. R Foundation for Statistical Computing Vienna, Austria

Roche-Lima A, Domaratzki M, Fristensky B (2014) Metabolic network prediction through pairwise rational kernels. BMC Bioinformat 15:318

Scholkopf B, Smola AJ (2002) Learning with kernels: support vector machines, regularization, optimization and beyond. MIT Press, Cambridge

Sikorski RS, Hieter P (1989) A system of shuttle vectors and yeast host strains designed for efficient manipulation of DNA in *Saccharomyces cerevisiae*. Genetics 122:19–27

Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D, Futcher B (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. Mol Biol Cell 9(3273):97

Tyree S, Gardner JR, Weinberger KQ, Agrawal K, Tran J (2014) Parallel support vector machines in practice. arXiv preprint arXiv:1404.1066

Vert J-P, Qiu J, Noble W (2007) A new pairwise kernel for biological network inference with support vector machines. BMC Bioinformat 8:S8

Yamanishi Y (2010) Supervised inference of metabolic networks from the integration of genomic data and chemical information. Elem Comput Syst Biol 8:189

Yamanishi Y, Vert JP (2007). Kernel matrix regression. arXiv preprint q-bio/0702054

Yamanishi Y, Vert J, Kanehisa M (2004) Protein network inference from multiple genomic data: a supervised approach. Bioinformatics 20:i363–i370

Yamanishi Y, Vert J, Kanehisa M (2005) Supervised enzyme network inference from the integration of genomic data and chemical information. Bioinformatics 21:i468–i477

Yu J, Guo M, Needham CJ, Huang Y, Cai L, Westhead DR (2010) Simple sequence-based kernels do not predict protein–protein interactions. Bioinformatics 26:2610–2614