Integration of heterogeneous data sources for gene function prediction using Decision Templates and ensembles of learning machines

Matteo Re[a,*], Giorgio Valentini[a,*]

[a]*Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano, Via Comelico 39, Italy*

Abstract

Several solutions have been proposed to exploit the availability of heterogeneous sources of biomolecular data for gene function prediction, but few attention has been dedicated to the evaluation of the potential improvement in functional classification results that could be achieved through data fusion realized by means of ensemble-based techniques. In this contribution we test the performance of several ensembles of Support Vector Machine (SVM) classifiers, in which each component learner has been trained on different types of bio-molecular data, and then combined to obtain a consensus prediction using different aggregation techniques. Experimental results using data obtained with different high-throughput biotechnologies show that simple ensemble methods outperform both learning machines trained on single homogeneous types of bio-molecular data, and vector space integration methods.

Key words: Majority voting; decision templates; decision fusion; data integration; gene function prediction.

1. Introduction

Functional classification of unannotated genes, and the improvement of the existing gene functional annotation catalogs, is a central problem in modern functional genomics and bioinformatics.

One of the main topics that characterize gene function prediction is the problem of the integration of multiple heterogeneous data sources. Indeed high-throughput biotechnologies make available increasing types and amount of biomolecular data, and several works pointed out that the integration of heterogeneous biomolecular data plays a central role to improve the accuracy of gene function prediction [1].

A first approach proposed in literature consists in modelling interactions between gene products using graphs and functional linkage networks [2]: integration is exploited through a "conjunctive method", i.e. by including exactly the edges that can be

[*] Corresponding author.
Email addresses: `re@dsi.unimi.it` (Matteo Re), `valentini@dsi.unimi.it` (Giorgio Valentini)

confirmed in each source graph [3], or by applying a probabilistic evidence integration scheme based on graphical models [4]. Another approach is based on a direct “vector-space integration” (VSI) by which different vectorial data are concatenated [5]. Kernel methods, by exploiting the closure property with respect to the sum, represents another valuable research direction for the integration of biomolecular data [6].

All these methods suffer of limitations and drawbacks, due to their limited scalability to multiple data sources (e.g. Kernel integration methods based on semidefinite programming [6]), to their limited modularity when new data sources sources are added (e.g. vector-space integration methods), or when data are available with different data type representations (e.g. functional linkage networks and vector-space integration).

A new possible approach is based on ensemble methods, but as observed by Noble and Ben-Hur, not much work has been done to apply classifier integration methods to protein function prediction [1]. To our knowledge, only few works have been proposed, such as the ”late integration” of kernels trained on different sources of data [7], or the Naive-Bayes integration of the outputs of SVMs in the context of the hierarchical classification of genes [8].

In this contribution we investigate the effectiveness of some simple ensemble methods based on majority voting and Decision Templates [9] in order to integrate heterogeneous biomolecular data sources for the prediction of gene functions.

2. Biomolecular data integration with ensemble methods and Decision Templates

2.1. Reasons for combining biomolecular data through ensembles

Apart from the general statistical, representational and computational reasons for combining multiple classifier systems [10], there are several reasons to apply ensemble methods in the specific context of genomic data fusion for gene function prediction.

At first, continuous advances in high-throughput biotechnologies provide new types of data, as well as updates of existing biomolecular data available for gene prediction. In this context, ensemble methods are well-suited to embed new types of data or to update existing ones by training only the base learners devoted to the newly added or updated data, without retraining the entire ensemble. Moreover most ensemble methods scale well with the number of the available data sources, and problems that characterize other data fusion approaches are thus avoided. Using vectorial data for different sources there is no bias in the integration of large and small or sparse and dense vectors. More in general diverse types of data (e.g. sequences, vectors, graphs) can be easily integrated, because with ensemble methods the integration is performed at decision level. Data fusion of heterogeneous biomolecular data sources can be effectively realized by means of ensemble systems composed by base learners trained on different datasets, and then combining their outputs to compute the consensus decision.

2.2. Decision Templates and ensembles for gene function prediction

In the context of gene function classification, we need to estimate of the reliability of the prediction [8]. To this end, we use SVMs, with probabilistic output obtained by applying a sigmoid fitting to their output [11]. Thus a trained base classifier computes a function $d_j : X \rightarrow [0, 1]$ that estimates the probability that a given example $\mathbf{x} \in X$ belongs to a specific class ω_j. An ensemble combines the outputs of n base learners, each trained on a different type of biomolecular data, using a suitable combining function g to compute the overall probability μ_j for a given class ω_j:

$$\mu_j(\mathbf{x}) = g(d_{1,j}(\mathbf{x}), \ldots, d_{n,j}(\mathbf{x})) \tag{1}$$

A simple way to integrate different biomolecular data sources is represented by the weighed linear combination rule:

$$\mu_j(\mathbf{x}) = \sum_{t=1}^{n} w_t d_{t,j}(\mathbf{x}) \tag{2}$$

The weights are usually computed using an estimate of the overall accuracy of the base learners, but for gene function prediction, where the functional classes are largely unbalanced (positive examples are largely less than negative ones), we choose the F-measure (the harmonic mean between precision and recall). We consider two different ways to compute the weights:

$$w_t^{\ell} = \frac{F_t}{\sum_{t=1}^{n} F_t} \qquad\qquad w_t^{log} \propto log \frac{F_t}{1 - F_t} \tag{3}$$

The w_t^{ℓ} weights are obtained by a linear combination of the F-measures, and w_t^{log} by a logarithmic transformation. Independently of the choice of the weights the decision $D_j(\mathbf{x})$ of the ensemble about the class ω_j is taken using the estimated probability μ_j (eq. 2):

$$D_j(\mathbf{x}) = \begin{cases} 1, & \text{if } \mu_j(\mathbf{x}) > 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where output 1 correspond to positive predictions for ω_j and 0 to negatives.

Certain types of biomolecular data can be informative for some functional classes, but uninformative for others. Hence it would be helpful to take into account whether certain types can be informative or not, depending on the class to be classified. To this end *Decision Templates* [9] can represent a valuable approach. The main idea behind decision templates consists in comparing a ”prototypical answer” of the ensemble for the examples of a given class (the template), to the current answer of the ensemble to a specific example whose class needs to be predicted (the decision profile).

More precisely, the decision profile DP($\mathbf{x}$) for an instance $\mathbf{x}$ is a matrix composed by the $d_{t,j}$ ∈[0,1] elements representing the support given by the t^{th} classifier to class ω_j. Decision templates DT_j are the averaged decision profiles obtained from $\mathbf{X}_j$, the set of training instances belonging to the class ω_j:

$$DT_j = \frac{1}{|\mathbf{X}_j|} \sum_{\mathbf{x} \in \mathbf{X}_j} DP(\mathbf{x}) \tag{5}$$

Given a test instance we first compute its decision profile and then we calculate the similarity $\mathcal{S}$ between $DP(\mathbf{x})$ and the decision template DT_j for each class ω_j, from a set of c classes. As similarity measure the Euclidean distance is usually applied:

$$\mathcal{S}_j(\mathbf{x}) = 1 - \frac{1}{n \times c} \sum_{t=1}^{n} \sum_{k=1}^{c} [DT_j(t,k) - d_{t,k}(\mathbf{x})]^2 \tag{6}$$

The final decision of the ensemble is taken by assigning a test instance to a class with the largest similarity:

$$D(\mathbf{x}) = \arg\max_j \mathcal{S}_j(\mathbf{x}) \tag{7}$$

In our experimental setting we consider dichotomic problems, because a gene may belong or not to a given functional class, thus obtaining two-columns decision template matrices.

It is easy to see that with dichotomic problems the similarity ($\mathcal{S}_1$) (eq. 6) for the positive class and the similarity ($\mathcal{S}_2$) for the negative class become:

$$\mathcal{S}_1(\mathbf{x}) = 1 - \frac{1}{n} \sum_{t=1}^{n} [DT_1(t,1) - d_{t,1}(\mathbf{x})]^2 \tag{8}$$

$$\mathcal{S}_2(\mathbf{x}) = 1 - \frac{1}{n} \sum_{t=1}^{n} [DT_2(t,1) - d_{t,1}(\mathbf{x})]^2 \tag{9}$$

where DT_1 is the decision template for the positive class and DT_2 for the negative one. The final decision of the ensemble for a given functional class is:

$$D(\mathbf{x}) = \arg\max_{\{1,2\}} (\mathcal{S}_1(\mathbf{x}), \mathcal{S}_2(\mathbf{x})) \tag{10}$$

3. Experimental setup

We chose to perform our experiments using data collected for S. *cerevisiae* because it is among the most studied and well characterized model organisms and because of the great amount of biomolecular data available for this species.

We used protein-protein interaction data collected from BioGrid [12] , a database of protein and genetic interactions and from STRING [13], a collection of protein functional interactions inferred from heterogeneous data sources comprising, among the others, experimental data and information found in literature. Moreover, we considered homology relationships data using pairwise Smith-Waterman $\log E$ values between all pairs of yeast protein sequences. We included also protein domain data available from *Pfam* [14]. We considered the presence/absence of a particular protein domain in the proteins encoded by genes comprised in the dataset and the E-value assigned to each gene product by a collection of profile-HMMs each of which trained on a specific domain family. The E-values have been computed through HMMR software toolkit (`http://hmmer.janelia.org`). Finally we included into our experiment a dataset obtained by the integration of microarray hybridization experiments published in [15] [16]. The main characteristics of the data sets used in the experiments are summarized in Tab. 1.

Table 1: Datasets

Code	Dataset	examples	features	description
D_{ppi1}	PPI - STRING	2338	2559	protein-protein interaction data from [13]
D_{ppi2}	PPI - BioGRID	4531	5367	protein-protein interaction data from the *BioGRID* database [12]
D_{pfam1}	Protein domain log-E	3529	5724	Pfam protein domains with log E-values computed by the *HMMER* software toolkit
D_{pfam2}	Protein domain binary	3529	4950	protein domains obtained from *Pfam* database [14]
D_{expr}	Gene expression	4532	250	merged data of Spellman and Gasch experiments [15] [16]
D_{seq}	Pairwise similarity	3527	6349	Smith and Waterman log-E values between all pairs of yeast sequences

Table 2: FunCat classes

Code	**Description**	**Code**	**Description**
01	Metabolism	20	Cellular transport and transport routes
02	Energy	30	Cellular communication/ Signal transduction mechanism
10	Cell cycle and DNA processing		
11	Transcription	32	Cell rescue, defense and virulence
12	Protein synthesis	34	Interaction with the environment
14	Protein fate	40	Cell fate
16	Protein with binding function or cofactor requirement	42	Biogenesis of cellular components
18	Regulation of metabolism and protein function	43	Cell type differentiation

We considered yeast genes common to all data sets (about 1900), and we associated them to functional classes using the functional annotations of the Functional Catalogue (FunCat) database (version 2.1) [17].

In order to reduce the number of classification tasks required by the experimental setting we considered only the first level of the hierarchy of FunCat classes. In other words, we selected the roots of the trees of the FunCat forest (that is the most general and wide functional classes of the overall taxonomy). We also removed from the list of the target functional classes all those represented by less than 20 genes. This corresponds to restrict our classifications to only 15 FunCat classes (Tab. 2)

Each dataset was split into a training set and a test set (composed, respectively, by the 70% and 30% of the available samples). We performed a 3-fold stratified cross-validation on the training data for model selection: we computed the F-measure across folds, while varying the parameter σ of the gaussian kernel and the C regularization term, each ranging from 10^{-5} to 10^{5}. The optimal parameters selected through the internal cross-validation procedure described above have been used to train the final model on all the data available in the training set. All the experiments have been performed using the e1071 interface to LIBSVM and in house written R language scripts. To evaluate the performance on the separated test set we computed both the F-measure and the AUC (Area Under the ROC Curve). This choice is motivated by the large unbalance between positive and negative examples that characterizes gene function prediction problems: indeed on the average only a small subset of the available genes is annotated to each functional class. We compared the performances of single gaussian SVMs trained on each data set with those obtained with vector-space-integration

Table 3: Ensembles of learning machines: averages across the performed learning tasks of the F-measure, precision, recall and AUC (Area Under the Curve) computed on the test sets.

Metric	E_{lin}	E_{log}	E_{dt}	VSI	D_{avg}	D_{ppi2}
F	0.4347	0.4111	0.5302	0.3213	0.3544	0.4818
rec	0.3304	0.2974	0.4446	0.2260	0.2859	0.3970
prec	0.8179	0.8443	0.7034	0.6530	0.5823	0.6157
AUC	0.8642	0.8653	0.8613	0.7238	0.7265	0.8170

(VSI) techniques (using a linear SVM for classifier), and with the ensembles described in Sect. 2.2. VSI, known also as "early integration", is a data integration method by which vectors of different data sets are concatenated and used to directly train a learning machine [7]. We normalized the data with respect to the mean and standard deviation, separately for each data set.

4. Results

The summary of the results are reported in Tab. 3. The table shows the average F-measure, recall, precision and AUC across the 15 selected FunCat classes, obtained through the evaluation of the test sets (each constituted by 570 genes). The three first columns refer respectively to the weighted linear, logarithmic linear and decision template ensembles; VSI stands for vector space integration (Sect. 3), D_{avg} represents the averaged results of the single SVMs across the six datasets, and D_{ppi2} represents the single SVM that achieved the best performance, i.e. the one trained using protein-protein interactions data collected from BioGrid (Tab. 1). Tab. 4 shows the same results obtained by each single SVM trained on a specific biomolecular data set.

Table 4: Single SVMs: averages across the performed learning tasks of the F-measure, precision, recall and AUC (Area Under the Curve) computed on the test sets. Each SVM is identified by the same name of the data set used for its training (Tab. 1).

Metric	D_{ppi1}	D_{ppi2}	D_{pfam1}	D_{pfam2}	D_{expr}	D_{seq}
F	0.3655	0.4818	0.2363	0.3391	0.2098	0.4493
rec	0.2716	0.3970	0.1457	0.2417	0.1571	0.5019
prec	0.6157	0.6785	0.7154	0.6752	0.3922	0.4162
AUC	0.7501	0.8170	0.6952	0.6995	0.6507	0.7469

Looking at the values presented in Tab. 3, we see that, on the average, data integration through ensemble methods provides better results than single SVMs and VSI, independently of the applied combination rule. In particular, Decision Templates achieved the best average F-measure, while the average AUC is about equal for all the ensemble methods, but larger with respect to both single SVMs and VSI. Precision of the ensembles is relatively high: this is of paramount importance to drive the biological

Table 5: Results of the non-parametric test based on Mann-Withney statistics to compare AUCs between ensembles, VSI and single SVMs. Each entry represents wins-ties-losses between the corresponding row and column. Left: Comparison between ensembles and VSI; Right: Comparison between ensembles and VSI with single SVMs.

	VSI	E_{log}	E_{lin}
E_{log}	13-2-0	-	-
E_{lin}	13-2-0	0-14-1	-
E_{dt}	13-2-0	1-13-1	1-11-3

	D_{ppi1}	D_{ppi2}	D_{pfam1}	D_{pfam2}	D_{expr}	D_{seq}
E_{lin}	11-4-0	4-11-0	15-0-0	14-1-0	15-0-0	13-2-0
E_{log}	11-4-0	4-11-0	15-0-0	14-1-0	15-0-0	13-2-0
E_{dt}	11-4-0	4-11-0	15-0-0	14-1-0	15-0-0	13-2-0
VSI	1-11-3	0-8-7	2-11-2	1-14-0	4-11-0	0-12-3

validation of ”in silico” predicted functional classes: considering the high costs of biological experiments, we need to obtain a high precision (and possibly recall) to be sure that positive predictions are actually true with the largest confidence.

To understand whether the differences between AUC scores in the 15 dichotomic tasks are significant, we applied a non parametric test based on the Mann-Withney statistic [18], using a recently proposed software implementation [19].

Tab. 5 shows that at 0.01 significance level in most cases there are no differences between AUC scores of the ensembles, while the difference is significant when we compare the ensembles with VSI, independently of the combination method (Tab. 5, left). It is worth noting that ensembles undergo no losses when compared with single SVMs (Tab. 5, right): we can safely choose any ensemble to obtain equal or better results than any of the single SVMs. On the contrary in many cases VSI shows worse results than single SVMs. Nevertheless, we can observe that a single SVM trained with Ppi-2 data achieves good results (11 ties with ensembles and an average AUC $\simeq$ 0.81 w.r.t. 0.86 of the ensembles, Tab. 3 and 5), showing that large protein-protein interactions data alone provide information sufficient to correctly predict several FunCat classes in S. *cerevisiae*, even if this is not necessarily true for different organisms.

An interesting observation is that the observed averaged AUCs of the base learners trained using PFAM data (D_{pfam1} and D_{pfam2}) are lower than the ones observed for the component classifiers trained on protein-protein interactions data (D_{ppi1} and D_{ppi2}). This seems to be in contrast with data previously published in [21] where the PFAM data were found to be the most informative source of information for Gene Ontology-based gene function prediction. This apparent discrepancy can be explained considering the different annotation policies adopted by the Gene Ontology [20] and FunCAT functional annotation projects, being the latter mainly based on experimental data [17], while the former on a broader set of supporting evidences (ranging from experimental observation to in-silico analyses). Being Protein-protein interactions experimental evidences they are expected to be more informative for the prediction of FunCAT annotations, while PFAM patterns could be, as pointed out in [21], of capital importance in order to achieve good prediction performances using a functional annotation scheme whose annotations are not required to be mainly of experimental origin.

A second, and more important, difference between the two experiments is that it is simpler to obtain a larger coverage in terms of PPI and, more in general, experimental evidences in the unicellular yeast model organism than in the multicellular Mus

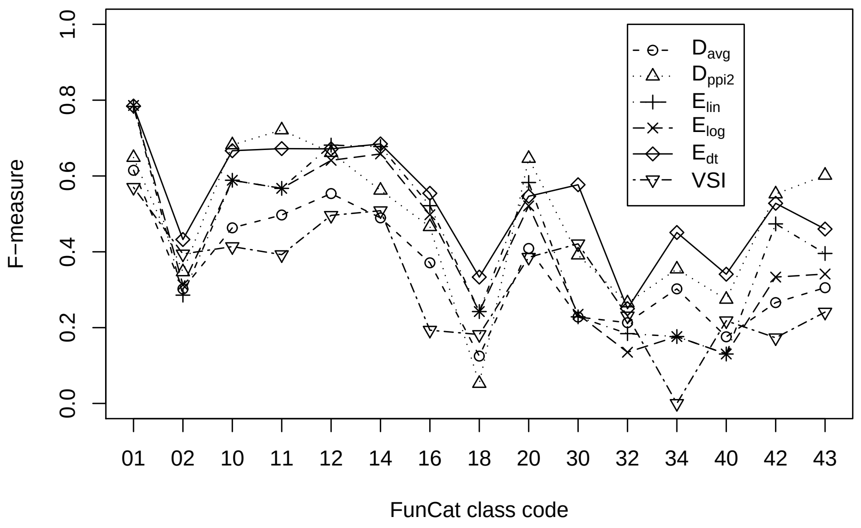

Figure 1: Comparison of the F-measures achieved in gene prediction: D_{avg} stands for the average across SVM single learners, D_{ppi2} for the best single SVM, E_{lin}, E_{log}, E_{dt} for weighted linear, logarithmic and decision template ensembles, VSI for vector space integration.

musculus. This is reflected by the different fractions of genes with at least one experimental functional annotation in S.*cerevisiae* (85.4%) and M.*musculus* (57.8%) in the Gene Ontology [22]. F-measure performances are summarized in Fig. 1: all ensemble methods outperform on the average single SVMs. Nevertheless the best single SVM (D_{ppi2}) outperforms weighted linear and logarithmic ensembles for some functional classes, but decision templates are in most cases better than the best single SVM, and significantly better than VSI.

5. Conclusions

In this work we investigated the impact on yeast genes functional classification performances of ensemble-based data fusion methods.

Our experiments demonstrated the potential benefits introduced by the usage of simple ensemble-based prediction systems for the integration of multiple sources of data in gene functional classification problems. The ensembles were able to outperform the averaged performances of base learners in all the gene function prediction tasks, achieving the best results in terms of AUC, and Decision Templates showed the best average F-measure across the 15 functional classes.

These results, obtained with relatively simple combining methods, show the effectiveness of the ensemble approach in the integration of heterogeneous biomolecular data sources for gene function prediction. Moreover we think that the application and the development of more refined ensemble methods, exploiting the modularity and scalability that characterizes the ensemble approach, represent a promising research line for gene function prediction using heterogeneous sources of complex biomolecular data.

Acknowledgments

The authors would like to thank the anonymous reviewers for their comments and suggestions, and gratefully acknowledge partial support by the PASCAL2 Network of Excellence under EC grant no. 216886. This publication only reflects the authors' views.

References

[1] W. Noble, A. Ben-Hur, Integating information for protein function prediction, in: T. Lengauer (Ed.), Bioinformatics - From Genomes to Therapies, Vol. 3, Wiley-VCH, 2007, pp. 1297–1314.

[2] U. Karaoz, et al., Whole-genome annotation by using evidence integration in functional-linkage networks, Proc. Natl Acad. Sci. USA 101 (2004) 2888–2893.

[3] E. Marcotte, M. Pellegrini, M. Thompson, T. Yeates, D. Eisenberg, A combined algorithm for genome-wide prediction of protein function, Nature 402 (1999) 83–86.

[4] O. Troyanskaya, et al., A Bayesian framework for combining heterogeneous data sources for gene function prediction (in *saccharomices cerevisiae*), Proc. Natl Acad. Sci. USA 100 (2003) 8348–8353.

[5] M. desJardins, P. Karp, M. Krummenacker, T. Lee, C. Ouzounis, Prediction of enzyme classification from protein sequence without the use of sequence similarity, in: Proc. of the 5th ISMB, AAAI Press, 1997, pp. 92–99.

[6] G. Lanckriet, T. De Bie, N. Cristianini, M. Jordan, W. Noble, A statistical framework for genomic data fusion, Bioinformatics 20 (2004) 2626–2635.

[7] P. Pavlidis, J. Weston, J. Cai, W. Noble, Learning gene functional classification from multiple data, J. Comput. Biol. 9 (2002) 401–411.

[8] Y. Guan, C. Myers, D. Hess, Z. Barutcuoglu, A. Caudy, O. Troyanskaya, Predicting gene function in a hierarchical context with an ensemble of classifiers, Genome Biology 9 (2008) S2.

[9] L. Kuncheva, J. Bezdek, R. Duin, Decision templates for multiple classifier fusion: an experimental comparison, Pattern Recognition 34 (2) (2001) 299–314.

[10] T. Dietterich, Ensemble methods in machine learning, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems. First International Workshop, MCS 2000, Cagliari, Italy, Vol. 1857 of Lecture Notes in Computer Science, Springer-Verlag, 2000, pp. 1–15.

[11] H. Lin, C. Lin, R. Weng, A note on Platt's probabilistic outputs for support vector machines, Machine Learning 68 (2007) 267–276.

[12] C. Stark, B. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, M. Tyers, BioGRID: a general repository for interaction datasets, Nucleic Acids Res. 34 (2006) D535–D539.

[13] C. vonMering, et al., STRING: a database of predicted functional associations between proteins., Nucleic Acids Research 31 (2003) 258–261.

[14] R. Finn, J. Tate, J. Mistry, P. Coggill, J. Sammut, H. Hotz, G. Ceric, K. Forslund, S. Eddy, E. Sonnhammer, A. Bateman, The Pfam protein families database, Nucleic Acids Research 36 (2008) D281–D288.

[15] P. Gasch, et al., Genomic expression programs in the response of yeast cells to environmental changes, Mol.Biol.Cell 11 (2000) 4241–4257.

[16] P. Spellman, et al., Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomices cerevisiae by microarray hybridization, Mol. Biol. Cell 9 (1998) 3273–3297.

[17] A. Ruepp, A. Zollner, D. Maier, K. Albermann, J. Hani, M. Mokrejs, I. Tetko, U. Guldener, G. Mannhaupt, M. Munsterkotter, H. Mewes, The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes, Nucleic Acids Research 32 (18) (2004) 5539–5545.

[18] E. Delong, D. Delong, D. Clarke-Pearson, Comparing the areas under two or more or more correlated Receiver Operating Characteristics Curves: a non parametric approach, Biometrics 44 (3) (1988) 837–845.

[19] I. Vergara, T. Norambuena, E. Ferrada, A. Slater, F. Melo, StAR: a simple tool for the statistical comparison of ROC curves, BMC Bioinformatics 9 (265) (2008).

[20] The Gene Ontology Consortium,Gene Ontology: tool for the unification of biology, Nat Genet. 25 (2000) 25–29.

[21] L. Pena-Castillo, et al., A critical assessment of Mus musculus gene function prediction using integrated genomic evidence, Genome Biology 9 (S2) (2008).

[22] S.Y. Rhee, et al., Use and misuse of the gene ontology annotations, Nat. Rev. genetics 9 (2008).