# Ensemble Based Data Fusion for Gene Function Prediction

Matteo Re and Giorgio Valentini

DSI, Dipartimento di Scienze dell' Informazione,
Università degli Studi di Milano,
Via Comelico 39, 20135 Milano, Italia
{re,valentini}@dsi.unimi.it

**Abstract.** The availability of an ever increasing amount of data sources due to recent advances in high throughput biotechnologies opens unprecedented opportunities for genome-wide gene function prediction. Several approaches to integrate heterogeneous sources of biomolecular data have been proposed in literature, but they suffer of drawbacks and limitations that we could in principle overcome by applying multiple classifier systems. In this work we evaluated the performances of three basic ensemble methods to integrate six different sources of high-dimensional biomolecular data. We also studied the performances resulting from the application of a simple greedy classifier selection scheme, and we finally repeated the entire experiment by introducing a feature filtering step. The experimental results show that data fusion realized by means of ensemble-based systems is a valuable research line for gene function prediction.

## 1 Introduction

The integration of multiple sources of heterogeneous biomolecular data is a key item for the prediction of gene function at genome-wide level. More in general, functional classification of unannotated genes is a central problem in modern functional genomics and bioinformatics [1]. The ever increasing amount of biomolecular data produced in last years as effect of recent advances in high-throughput biotechnologies did not result into a corresponding improvement in gene function prediction accuracy, because the additional complexity introduced by the need to integrate heterogeneous data sources constitutes a serious limiting factor [2]. To deal with this problem, several approaches have been proposed in literature. A first one is based on a direct "vector-space integration" by which different vectorial data are concatenated [3]. Modelling interactions between gene products using graphs and functional linkage networks is another valuable research line, as well as the application of probabilistic graphical models [4]. Kernel methods, by exploiting the closure property of kernels with respect to the sum and other algebraic operators, represent another interesting approach for the integration of biomolecular data [5]. Nevertheless, all these methods suffer of limitations and drawbacks, due to their limited scalability to multiple data

J.A. Benediktsson, J. Kittler, and F. Roli (Eds.): MCS 2009, LNCS 5519, pp. 448–457, 2009.

sources (i.e. Kernel integration methods based on semidefinite programming [5]), to their limited modularity when new data sources are added (i.e. vector-space integration methods), or when the available biomolecular data are characterized by different structural features (i.e. functional linkage networks and vector-space integration).

A new possible approach is represented by ensemble methods, but not much work has been done to apply classifier integration to gene function prediction [2]. To our knowledge, only few works have been proposed, such as the "late integration" of kernels trained on different sources of data [6], or the Naive-Bayes integration of the outputs of SVMs in the context of the hierarchical classification of genes [7]. Ensemble-based data fusion techniques have been successfully applied in several domains, ranging from biomedical applications [8] to the classification of multisource remote-sensing images [9]. However, there are several reasons to apply ensemble methods in the specific context of genomic data fusion for gene function prediction. At first, biomolecular data differing for their structural characteristics (e.g. sequences, vectors, graphs) can be easily integrated, because with ensemble methods the integration is performed at the decision level, combining the outputs produced by classifiers trained on different datasets. Moreover, as new types of biomolecular data, or updates of data contained in public databases, are made available to the research community, ensembles of learning machines are able to embed new data sources or to update existing ones by training only the base learners devoted to the newly added or updated data, without retraining the entire ensemble. Finally most ensemble methods scale well with the number of the available data sources, and problems related to the addition of newly available sources of biomolecular data can be easily managed.

In this contribution we investigate the effectiveness of different types of ensemble systems in gene function prediction. We also evaluate the effect on the quality of predictions due to the introduction of a simple base classifier selection scheme. We finally repeat the entire experiment introducing a feature selection step. The results are then compared with baseline methods to provide an overview of the potentialities of multiple classifier systems in gene function prediction.

## 2 Methods

In our experiments, to integrate different sources of biomolecular data, we chose relatively simple methods, such as weighted average combination methods and decision templates. As a second step we considered ensembles based on base learner selection, according to the test-and-select approach, and finally we applied ensembles combined with simple feature filtering methods to reduce the high dimensionality that characterize biomolecular data.

### 2.1 Ensemble Methods for Biomolecular Data Fusion

Data fusion can be realized by means of an ensemble system composed by learners trained on different "views" of the data and then combining the outputs of the component learners. Each type of data may capture different and complementary characteristics of the objects to be classified and the resulting ensemble may obtain better prediction capabilities through the diversity and the anti-correlation of the base learner responses.

In particular, each type of biomolecular data $B_1, B_2, \ldots, B_T$ is characterized by different features $f_1, f_2, \ldots, f_T$, where $T$ is the number of the available data sources. Thus, an example $x$ is characterized by different sets of features:

$$\mathbf{x} = < \mathbf{x}_{f_1}, \mathbf{x}_{f_2}, \ldots, \mathbf{x}_{f_T} > \tag{1}$$

where $\mathbf{x}_{f_t}$ represents the data relative to the features $f_t$ of a specific data set $B_t \subset X_t$.

A classifier trained on data $B_t$ computes a function $d_{t,j} : X_t \to [0, 1]$ that estimates the support (e.g. the probability) that a given example $x$ belongs to a specific class $\omega_j$. In our experiments we applied a sigmoid fitting to the output of SVMs, to obtain an estimate of the probability that a given example belongs to a given class [10]. An ensemble combines the outputs of $T$ base learners, each trained on a different type of biomolecular data, using a suitable combining function $g$ to compute the overall support $\mu_j$ for a given class $\omega_j$:

$$\mu_j(\mathbf{x}) = g(d_{1,j}(\mathbf{x}_{f_1}), d_{2,j}(\mathbf{x}_{f_2}), \ldots, d_{T,j}(\mathbf{x}_{f_T})) \tag{2}$$

At first, we combine the base classifiers through the classical *weighted average rule*:

$$\mu_j(\mathbf{x}) = \sum_{t=1}^{T} w_t d_{t,j}(\mathbf{x}_{f_t}) \tag{3}$$

In our experiments we computed the weights according to a convex combination rule ($w_t^c$) and a logarithmic transformation ($w_t^{log}$):

$$w_t^c = \frac{F_t}{\sum_{t=1}^{T} F_t} \qquad\qquad w_t^{log} \propto log \frac{F_t}{1 - F_t} \tag{4}$$

In both cases we use the F-measure $F_t$, i.e. the harmonic mean between precision and recall, instead of the classical accuracy, since the gene functional classes are largely unbalanced (positive examples are largely less than negative ones). $F_t$ measures are obtained by "internal“ cross-validation on the training data. The ensemble chooses the class $\omega_j$, according to the estimated probability $\mu_j$ (eq. 3):

$$D_j(\mathbf{x}) = \begin{cases} 1, & \text{if } \mu_j(\mathbf{x}) > h \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where output 1 corresponds to positive predictions for $\omega_j$ and 0 to negatives. A reasonable value for the threshold $h$ is 0.5 (if $\mu_j$ estimates probabilities). Note that in this setting an example $\mathbf{x}$ may belong to more than one class (eq. 5), thus modeling the multilabel classification problem that characterizes gene function prediction.

Some base learners trained on specific biomolecular data may incorrectly predict the examples for a given gene functional class for several reasons. For instance certain types of biomolecular data can be informative for some functional classes, but uninformative for others. In order to take into account systematic incorrect answers of certain base learners, *Decision Templates* [11] can represent a valuable approach. In this approach the decision profile DP($\mathbf{x}$) for an instance $\mathbf{x}$ is a matrix composed by the $d_{t,j}$ $\in$[0,1] elements representing the support given by the $t^{th}$ classifier to class $\omega_j$. Decision templates $DT_j$ are the averaged decision profiles obtained from $\mathbf{X}_j$, the set of training instances belonging to the class $\omega_j$:

$$DT_j = \frac{1}{|\mathbf{X}_j|} \sum_{\mathbf{x} \in \mathbf{X}_j} DP(\mathbf{x}) \tag{6}$$

The similarity $\mathcal{S}$ between the decision template $DT_j$ for a class $\omega_j$, $1 \leq j \leq C$, and the decision profile for a given test instance $\mathbf{x}$ is:

$$\mathcal{S}_j(\mathbf{x}) = 1 - \frac{1}{T \times C} \sum_{t=1}^{T} \sum_{k=1}^{C} [DT_j(t,k) - d_{t,k}(\mathbf{x})]^2 \tag{7}$$

and the final decision of the ensemble is computed by assigning the test instance to the class with the largest similarity:

$$D(\mathbf{x}) = \arg\max_j \mathcal{S}_j(\mathbf{x}) \tag{8}$$

For gene prediction we consider two-classes problems, because a gene may belong or not to a given functional class. To simplify the notation, we denote the positive class by 1 and the negative by 2. In this context, exploiting the fact that $d_{t,2}(\mathbf{x}) = 1 - d_{t,1}(\mathbf{x})$, the similarity $\mathcal{S}$ (eq. 7) for the positive and the negative class class becomes:

$$\mathcal{S}_1(\mathbf{x}) = 1 - \frac{1}{T} \sum_{t=1}^{T} [DT_1(t,1) - d_{t,1}(\mathbf{x})]^2 \tag{9}$$

$$\mathcal{S}_2(\mathbf{x}) = 1 - \frac{1}{T} \sum_{t=1}^{T} [DT_2(t,1) - d_{t,1}(\mathbf{x})]^2 \tag{10}$$

and the final decision of the ensemble is:

$$D(\mathbf{x}) = \arg\max(\mathcal{S}_1(\mathbf{x}), \mathcal{S}_2(\mathbf{x})) \tag{11}$$

### 2.2 Feature Filtering

Feature selection methods can select the most significant features and can reduce the high dimensionality that characterize most biomolecular data.

To reduce the computational complexity we introduce a simple filtering method based on the t-test statistic: More precisely, we applied the two-sample

Welch t-test to verify the null hypothesis $\mathcal{H}_j$ of no difference between the means of feature values of the two given positive and negative sets of genes at a given significance level $\alpha$. Since the number of features for each data set is in the order of thousands, we need to restate the problem in a multiple hypothesis test setting. In particular we applied the Benjamini and Hochberg (BH) [12] procedure to control the false discovery rate *FDR* (that is the expected proportion of false positives among the rejected hypotheses). This procedure is applied separately for each data set.

### 2.3 Base Learner Selection

According to the *test and select* methodology [13], we apply a variant of the "choose the best" technique [14] to select a subset of "optimal" classifiers. More precisely we select the "best" subset of base classifiers (each one trained on a different source of biomolecular data) according to the F-measure estimated by internal cross-validation on the the training set. A high level scheme of the adopted "test and select" procedure is reported below:

1. Separately for each available data, select the most significant features using the two-sample t-test with Benjamini and Hochberg p-value correction (Sect. 2.2).
2. Train the base learners on the heterogeneous data sets filtered according to step 1.
3. Select the $n$ learners with the best F-measure estimated by internal cross-validation on the training set
4. Evaluate the ensembles with the $n$ best learners on a separated test set.

We applied the "test and select" procedure with and without the first step (feature filtering with "corrected" t-test). Note that at step 2 and 3 a base learner model selection can also be performed using cross-validation on the training data. Weighted average rule and decision templates (Sect. 2.1) are the aggregation strategies adopted to combine the output of the base learners.

## 3 Experimental Setup

We collected several sources of biomolecular data to classify genes of the yeast, an eukaryotic unicellular model organism. In particular we used protein-protein interaction data collected from BioGrid [15] and STRING [16], a collection of physical and genetic interactions obtained from different types of biological experiments and from literature. Moreover we included data to register the presence/absence of a particular protein domain in the proteins encoded by genes comprised in the dataset [17] and the E-value assigned to each gene product to a collection of profile-HMMs computed through the HMMR software toolkit (`http://hmmer.janelia.org` ). We considered also homology relationships data using pairwise Smith-Waterman $\log E$ values between all pairs of yeast sequences. Finally we included into our experiment a dataset obtained by the

**Table 1.** Datasets

| Code | Dataset | examples | features | description |
|---|---|---|---|---|
| D1 | Protein domain binary | 3529 | 4950 | protein domains obtained from *Pfam* database [17] |
| D2 | Protein domain log-E | 3529 | 5724 | Pfam protein domains with log E-values computed by the *HMMER* software toolkit |
| D3 | Gene expression | 4532 | 250 | merged data of Spellman and Gasch experiments [18] [19] |
| D4 | PPI - BioGRID | 4531 | 5367 | protein-protein interaction data from the *BioGRID* database [15] |
| D5 | PPI - STRING | 2338 | 2559 | protein-protein interaction data from [16] |
| D6 | Pairwise similarity | 3527 | 6349 | Smith and Waterman log-E values between all pairs of yeast sequences |

integration of microarray hybridization experiments published in [18] [19]. The main characteristics of the data sets used in the experiments are summarized in Tab. 1. The genes represented in the datasets under investigation have been associated to functional classes using the functional annotations collected in the Functional Catalogue (FunCat) database version (2.1) [20].

In our experiments we considered only the first level of the hierarchy of FunCat classes, that is the most general and wide 15 functional classes of the overall taxonomy.

We considered the intersection between all the datasets, resulting into a final collection of 1910 yeast genes. In other words we used in our experiments only the genes for which experimental measures were available for all the types of data. Each resulting dataset was randomly 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, using gaussian SVMs as base learners. We chose the F-measure for both model selection and to evaluate the performances on the separated test set, because most FunCat classes are unbalanced, with positive examples largely lower than negatives.

We then applied a test and select procedure, by choosing the best 2, 3 or 4 classifiers according to the F-measure evaluated by cross-validation on the training set (Sect.2.3). The test and select procedure has been applied with and without feature selection according to a two-sample t-test and a Benjamini and Hochberg correction at 0.05 significance level (Sect.2.2).

## 4 Results

Tab. 2 summarizes the averages across the performed 15 dichotomic learning tasks of the F-measure, recall, precision and specificity computed on the test sets using respectively:

1. The ensemble methods described in Sect. 2.1 using all the available data sets and base learners
2. The test-and-select procedure outlined in Sect. 2.3.
3. The feature filtering step added before the test-and-select procedure (Sect. 2.2).

**Table 2.** Summary of ensemble results. $L_{best}$ refers to the best single learner, $L_{avg}$ to the average results of single SVMs; $E_{lin}$ and $E_{log}$ to weighted average combination with respectively linear and logarithmic weights; $E_{DT}$ stands for decision templates ensembles.

| A) | **Results using all the available base learners** | | | | |
|---|---|---|---|---|---|
| Metric | $L_{best}$ | $L_{avg}$ | $E_{lin}$ | $E_{log}$ | $E_{DT}$ |
| F | 0.4816 | 0.3470 | 0.4403 | 0.4112 | 0.5302 |
| rec | 0.3970 | 0.2859 | 0.3304 | 0.2974 | 0.4446 |
| prec | 0.6785 | 0.5823 | 0.8179 | 0.8443 | 0.7034 |
| spec | 0.9516 | 0.9533 | 0.9798 | 0.9850 | 0.9594 |

| B) | **Results with test and select procedures** | | | | |
|---|---|---|---|---|---|
| Metric | $L_{best}$ | $L_{avg}$ | $E_{lin}$ | $E_{log}$ | $E_{DT}$ |
| F | 0.4816 | 0.3470 | 0.5436 | 0.5441 | 0.5698 |
| rec | 0.3970 | 0.2859 | 0.4793 | 0.4778 | 0.5164 |
| prec | 0.6785 | 0.5823 | 0.6723 | 0.6591 | 0.6435 |
| spec | 0.9516 | 0.9533 | 0.9538 | 0.9573 | 0.9447 |

| C) | **Results with test and select and feature filtering** | | | | |
|---|---|---|---|---|---|
| Metric | $L_{best}$ | $L_{avg}$ | $E_{lin}$ | $E_{log}$ | $E_{DT}$ |
| F | 0.4893 | 0.2638 | 0.5175 | 0.4912 | 0.6310 |
| rec | 0.3841 | 0.1927 | 0.3987 | 0.3711 | 0.5667 |
| prec | 0.7278 | 0.6141 | 0.8708 | 0.9042 | 0.7439 |
| spec | 0.9639 | 0.9775 | 0.9841 | 0.9871 | 0.9552 |

$L_{best}$ refers to the best single learner (trained on the D2 protein domain data set, Tab. 1), and $L_{avg}$ to the average results of the single SVMs across all the 6 data sets.

As reported in Tab. 2 A), the performances averaged across all the performed learning tasks are increased by the basic ensemble-based data fusion approaches involving the combination of all the component classifiers. The investigated combination strategies are able, on the average, to outperform the single learners. In particular the Decision Template combiner outperforms the single best classifier in the evaluation of the test set. The simple greedy strategy to test and select the "best" base learners for each classification task significantly enhances the performances of weighted average combination methods (from 0.41 to 0.54 with $E_{log}$), but also Decision templates gain from this approach (Tab. 2 B). By adding a simple feature selection step to the test and select methods we can observe that only Decision templates are able to improve their performances (Tab. 2 C). Indeed on the average the performances of single learners largely decrease (the F-measure falls from from 0.34 to 0.26), as well as the performances of weighted average ensembles, even if the relative decrement of the latter is lower. In all cases, independently of the adopted ensemble method and with or without feature selection, the ensembles of learning machines largely

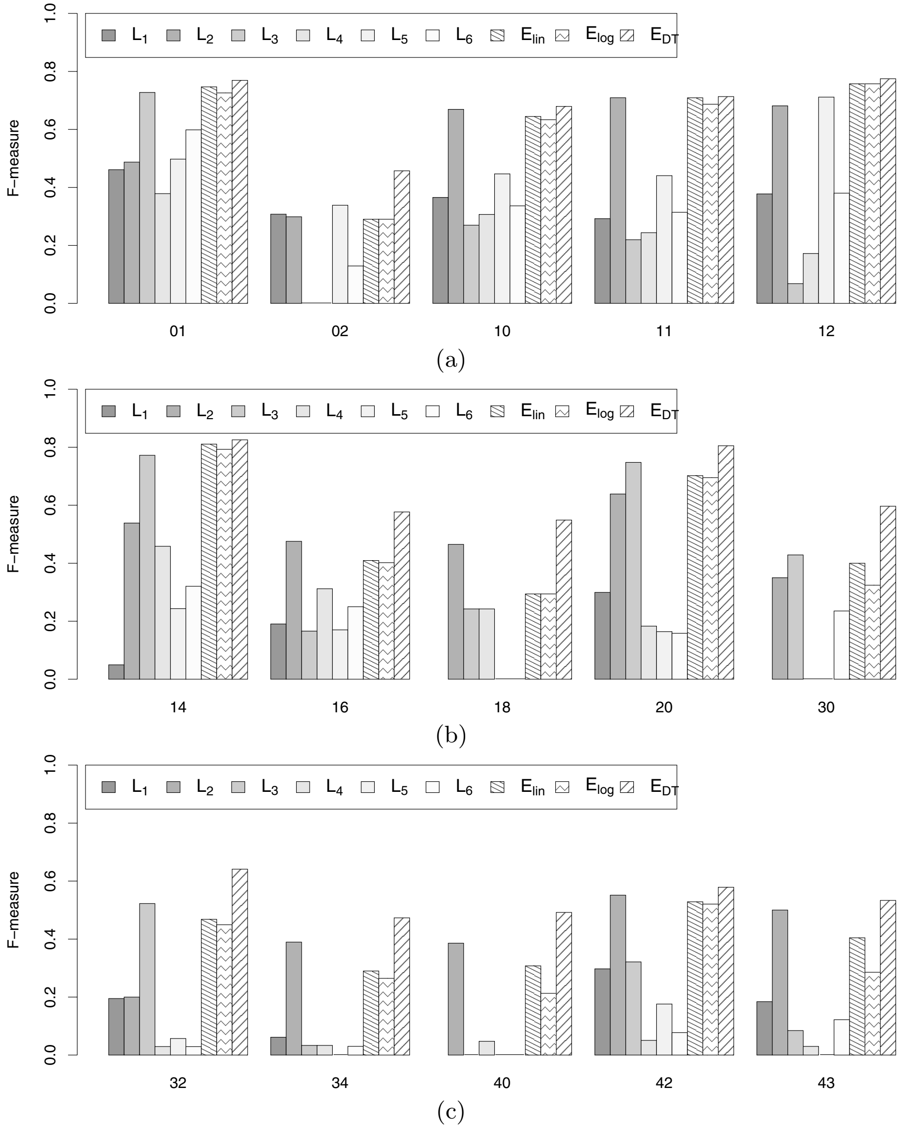

**Fig. 1.** Per class F-measure results of ensemble methods with base learner selection and feature filtering. For each FunCat class, the first six shaded gray bars refer to single learners with feature filtering (from $L_1$ to $L_6$); the last three bars (filled with patterns) correspond respectively to weighted average combination with linear ($E_{lin}$) and logarithmic ($E_{log}$) weights and decision template ($E_{DT}$) ensembles. a) Funcat classes 01, 02, 10, 11, 12; b) 14, 16, 18, 20, 30; c) 32, 34, 40, 42, 43.

outperform the average results of the single SVMs. Moreover in most cases ensemble methods outperform also the best single SVM, and in particular decision templates obtain better results than the best single SVM on all the gene function prediction tasks (Fig. 1). It is worth noting that ensemble methods achieve a very high precision (Tab. 2): this is of paramount importance to drive the biological validation of novel predicted genes whose function is unknown or only partially known, in order to reduce the costs of possible false positives.

Each type of biomolecular data set captures different characteristics of genes, and can be informative for some classes but uninformative for the prediction of other classes of genes. From this standpoint we can understand the reasons why simple decision fusion techniques may improve gene function prediction. In particular decision templates seem to better exploit the different characteristics of the available source of biomolecular data. Indeed, through the decision templates, also relatively uncertain or wrong responses of base learners can provide useful information for the decision of the ensemble, especially if this behaviour is consistently maintained across the data. This is confirmed also by the fact that test and select methods with decision templates to combine the output of the selected base learners require on the average more learners than weighted average ensembles (data not shown). Decision templates are thus able to exploit also the characteristics of the less informative base learners to improve the predictions of the overall ensemble.

## 5 Conclusions

In this work we investigated the effectiveness of ensemble-based data fusion methods on the functional classification of yeast genes. The ensembles are able to outperform the averaged performances of single SVMs in all the gene function prediction tasks, achieving the best results in terms of precision and recall. The performances are further improved by a simple "choose the best" selection strategy, and a feature filtering method is able to enhance the results of decision templates. Considering the F-measure that summarizes both precision and recall, the experimental results show that data fusion realized by means of ensemble systems is a valuable research line in gene function prediction and that Decision Templates may represent a good choice for biomolecular data integration.

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

[2] Noble, W., Ben-Hur, A.: Integating information for protein function prediction. In: Bioinformatics - From Genomes to Therapies, pp. 1297–1314. Wiley, Chichester (2007)

[3] des Jardins, M., et al.: Prediction of enzyme classification from protein sequence without the use of sequence similarity. In: Proc. of the 5th ISMB, pp. 92–99 (1997)

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

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

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

[7] Guan, Y., et al.: Predicting gene function in a hierarchical context with an ensemble of classifiers. Genome Biology 9 (2008)

[8] Polikar, R., et al.: An ensemble based data fusion approach for early diagnosis of Alzheimer disease. Information Fusion 9, 83–95 (2008)

[9] Benediktsson, J., Chanussot, J., Fauvel, M.: Multiple classifier systems in remote sensing: From basics to recent developments. In: Haindl, M., Kittler, J., Roli, F. (eds.) MCS 2007. LNCS, vol. 4472, pp. 501–512. Springer, Heidelberg (2007)

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

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

[12] Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. R. Statist. Soc. B 57, 289–300 (1995)

[13] Roli, F., Giacinto, G., Vernazza, G.: Methods for Designing Multiple Classifier Systems. In: Kittler, J., Roli, F. (eds.) MCS 2001. LNCS, vol. 2096, pp. 78–87. Springer, Heidelberg (2001)

[14] Partridge, D., Yates, W.: Engineering multiversion neural-net systems. Neural Computation 8, 869–893 (1996)

[15] Stark, C., et al.: BioGRID: a general repository for interaction datasets. Nucl. Acids Res. 34, D535–D539 (2006)

[16] von Mering, C., et al.: STRING: a database of predicted functional associations between proteins. Nucl. Acids Res. 31, 258–261 (2003)

[17] Finn, R., et al.: The Pfam protein families database. Nucl. Acids Res. 36, 281–288 (2008)

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

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

[20] Ruepp, A., et al.: The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. Nucl. Acids Res. 32, 5539–5545 (2004)