Bias and benefit induced by intra-species homologies in guilt by association methods to predict protein function

Laurent Brehelin, Olivier Gascuel

Bias and benefit induced by intra-species homologies in guilt by association methods to predict protein function

Laurent Bréhélin and Olivier Gascuel

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier , UMR CNRS 5506,
161 rue Ada, 34392 MONTPELLIER Cedex 5, France
brehelin@lirmm.fr gascuel@lirmm.fr

Abstract: *The guilt by association (GBA) principle is used in several supervised and non-supervised methods to functionally annotate uncharacterized genes from transcriptomic data or from other information source. However, these methods do not distinguish between genes which have or have not intra-species homologues. We show here that functional annotation and intra-species homology are strongly dependent. We emphasize that applying any GBA method not accounting for this form of homology has two opposite effects: it leads to over-estimating the method performance on the genes with no intra-species homologues, and to losing the benefit of homology on the other genes. Bias and benefit are measured on P. falciparum and Yeast, and a general scheme to properly apply the GBA principle is given. All together, this method improves over previous standard applications of the GBA principle.*

Keywords: Protein Function Prediction, Guilt By Association, Homology, Transcriptome, Gene Ontology, *P. falciparum*, Yeast.

1 Introduction

A common principle known as *guilt by association* (GBA) states that genes with similar transcriptomic profile are likely to share common functional roles [10, 16]. The GBA principle has been used to help with gene functional annotation in several organisms [23, 14, 21, 24]. In most of these works, authors proceed in a non-supervised way: given a selected group of genes of similar transcriptomic profiles —*e.g.* obtained from a clustering algorithm [10, 15, 13]—, a statistical test is applied to reveal over-represented types of functions among the characterized genes, thus providing functional clues for the uncharacterized ones. For each potential type of function, the proportion of annotations in the set of selected genes is compared with that of a reference set (generally the complete set of genes on the microarray). Recently, several methods and software have been published to help with such analysis [8, 4, 20, 1, 2, 5, 18, 17].

A few other methods [6, 19] use the GBA principle in a supervised framework: first, using the transcriptome of the genes already characterized, a prediction function is built by a supervised learning algorithm; next this predictor is used to propose one or several type of functions to the uncharacterized genes. Moreover, a cross-validation (CV) procedure is run on the characterized genes. This procedure allows estimating the confidence level of the predictions with uncharacterized genes.

The GBA principle also apply to non-transcriptomic data. For example, interactome has been used for this purpose, since it has been suggested that proteins that share common interactors are likely to share common functions [7, 22, 9].

However, these approaches do not distinguish between genes which have, or have not, intra-species homologues. Homology is a well known source of bias in the protein structure (secondary and tertiary) prediction community. On the one hand, predicting the structure of a protein that has an homologue of known structure is by far more easy than when no homologue is known, since homologous proteins are likely to share the same structure. On the other hand, assessing a protein structure prediction method on a dataset that contains both homologous and non-homologous proteins leads to optimistically biased performance estimate. Here we show that intra-species homologies also affect the functional annotation methods based on GBA. Supervised and non-supervised methods are differently affected. For both types, the impact is evaluated on two organisms, and a method to deal with intra-species homologies is proposed. As we shall see, function prediction may benefit or get worse, depending on the presence or absence of intra-species homologies.

2 Data, definitions, notations

The study is based on two organisms: *P. falciparum* and *S. cerevisiae*. The transcriptomic data come from Le Roch et al. [14] and Gasch et al. [11], respectively. Gene functions are described by GO terms, and the annotations are those relative to the *Biological Process* ontology published on the GO website[1]. Only genes with precise enough annotations are used, *i.e.* genes that are only annotated with GO terms of high (> 0.3) prior probability are not considered here. We define the prior probability of a term as the number of characterized genes that are annotated by this term, divided by the total number of characterized genes of the organism.

In the following, two genes g_i and g_j are considered as homologous if the E-value associated with the `blastP` alignment of g_i over g_j or of g_j over g_i is smaller than 10^{-10}. The homology relation is denoted by the symbol $\|$. This is a symmetric relation, *i.e.* we have $g_i \parallel g_j \Rightarrow g_j \parallel g_i$. We use the closure of this relation —*i.e.* $g_i \parallel g_j$ and $g_j \parallel g_k \Rightarrow g_i \parallel g_k$— and compute that way clusters (cliques) of homologous genes. Co-expression is defined in a similar way: two genes are considered as co-expressed if the Pearson correlation coefficient of their profile is larger than 0.8.The Table 1 summarizes some statistics about the number of characterized and non-characterized genes that possess homologous or co-expressed characterized genes, in the two datasets.

We denote as $\mathbf{X}_i$ the set of GO terms associated with a characterized gene g_i. Let g_i and g_j be two genes annotated in GO. We consider that g_i and g_j have similar functions if they share several GO terms. We define the *sharing score* of two sets of terms X and Y as

$$S_{XY} = 1/2\frac{|\mathbf{X} \cap \mathbf{Y}|}{|\mathbf{X}|} + 1/2\frac{|\mathbf{X} \cap \mathbf{Y}|}{|\mathbf{Y}|}. \quad (1)$$

Hence, set pairs that share many terms have sharing score around 1, while set pairs with very different terms have sharing score near 0. Next, we define the *functional similarity* (FS) of two genes g_i and g_j as the sharing score of $\mathbf{X}_i$ and $\mathbf{X}_j$.

3 Intra-species homologies and functional annotations are strongly dependent

We computed the FS histograms of the co-expressed and homologous pairs of the two datasets. Figure 1 summarizes the results. For comparison purpose, we also computed the FS histograms of randomly composed pairs. The three distributions are very different. For both organisms, homologous gene pairs

[1] http://www.geneontology.org/

	all		hom.		co-ex.		co-ex. hom.	
	#C.	#N.	#C	#N	#C	#N	#C	#N
P. falciparum	1266	3893	670 (**53)**	996 (**26)**	1177 (**93)**	3551 (**91)**	306 (**24)**	188 (**5)**
S. cerevisiae	4209	1943	2009 (**48)**	339 (**17)**	1091(**26)**	372 (**19)**	214 (**5)**	23 (**1)**

Table 1. Number of characterized (#C) and non-characterized (#N) genes in the complete set of genes (column all), and in the sets of genes with: (i) at least one characterized homologue (column hom.); (ii) at least one characterized co-expressed gene (column co-ex.); (iii) at least one characterized co-expressed homologue (column co-ex. hom.). Bold numbers indicate the proportion (in percent) of the complete set of characterized or non-characterized genes represented by each number. For example, 53% of characterized genes have at least one characterized homologue in *P. falciparum*.

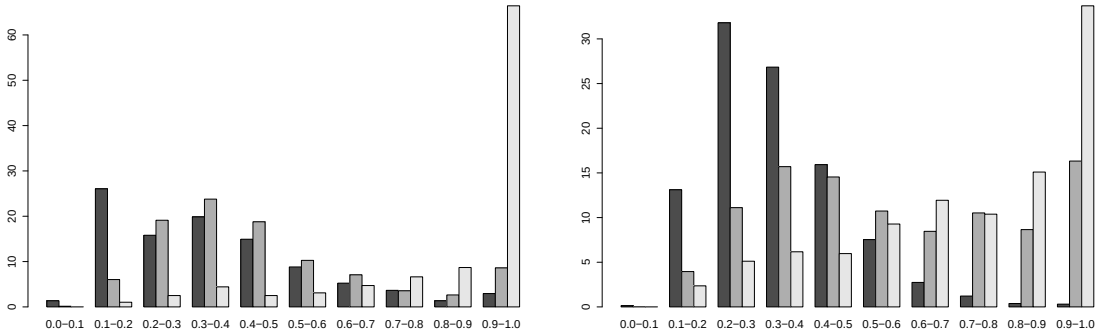

Figure 1. FS histograms (in percent) of random pairs (black), co-expressed pairs (grey), and homologous pairs (light grey), in *P. falciparum* (left) and *S. cerevisiae* (right).

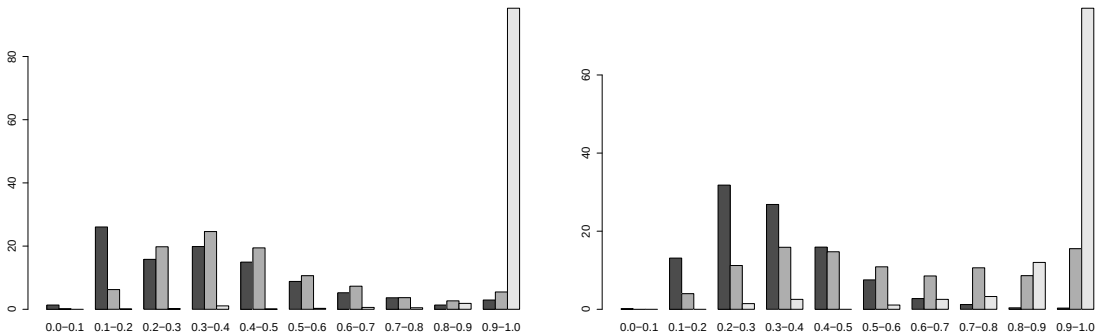

Figure 2. FS histograms (in percent) of co-expressed pairs which are homologous (light grey) or non-homologous (grey), and of random pairs (black), in *P. falciparum* (left) and *S. cerevisiae* (right).

have much higher probability to share the same GO terms than randomly paired genes. This is also true for co-expressed genes, but to a lesser extent. The phenomenon is even more acute when computing the FS of the co-expressed pairs and differentiating the homologous from the non-homologous pairs (see Figure 2). From these histograms we see that homologous co-expressed genes share almost always the same annotations. This could be seen as surprising as homologous genes within the same species are paralogues and are commonly thought to differ in terms of function. But GO annotations just draw large functional categories, and genes with high FS values can still differ when looking into the details of their functions. However, for non-homologous co-expressed genes, the situation is much less favorable; they differ just slightly from random pairs and function prediction from transcriptome only is expected to be an hard task.

This study shows that, as for protein structure prediction, it is important to differentiate between homologous and non-homologous genes when predicting functional annotations with the GBA principle. Genes that possess an homologous characterized gene are more easy to annotate and hence should be addressed with a specific procedure to fully benefit from this feature. Such a procedure is presented in the last section of the paper. Moreover, the need of processing separately genes with and without characterized homologues is reinforced by the difference in the proportions of genes that possess characterized homologues, in the set of characterized genes and in the set of non-characterized genes (see Table 1). Indeed, while around 50% of characterized genes possess a characterized homologue (both for the *P. falciparum* and *S. cerevisiae* datasets), this proportion falls around 20-25% for the uncharacterized genes. While this disproportion is not surprising —it comes from the fact that genomic similarity is widely used in the standard functional annotation procedures— it bias the performance estimate of the supervised and non-supervised methods, and thus induces erroneous annotations. The following sections detail this point in the two frameworks.

4 Bias in non-supervised methods

Let C be a cluster of co-expressed genes. If several characterized homologues are present in C, then a GO term t that annotates this group is likely to appear over-represented compared to the reference set. However, this over-representation does not constitute a functional clue for most of the uncharacterized genes, since the majority of these ones are not homologous to the group of characterized homologues. Actually, what the statistical test reveals is less the over-representation of t than the over-representation of a class of homologous genes in C.

Assume that the uncharacterized genes with characterized homologues have been processed separately. The last section of this paper details how this can be done, but note that only 20-25% of the uncharacterized genes are involved. We are now interested in the functional annotations of all the other uncharacterized genes. How can we apply the statistical test without bias? A simple solution involves selecting among the characterized genes and removing all but one gene for each class of homology in both the C cluster and the reference set, before running the statistical test. That way, characterized genes within C are no more homologous, just as the uncharacterized genes we aim to predict.

In order to evaluate the bias induced by the intra-species homologies when annotating the genes without characterized homologues, we apply the above correction procedure to the two datasets, and compared the results with those achieved without correction. We used the GOSTAT[2] software of Beissbarth and Speed [2] for the tests. More precisely, we used the GOSTAT2 version, with the False

[2] http://gostat.wehi.edu.au/

Discovery Rate procedure of Benjamini [3] to correct for multiple testing. Given two sets of genes, GOSTAT2 compares the proportions of every potential GO term in the two sets, and outputs the terms that are statistically over-represented in the first one. For the *P. falciparum* dataset, we used the gene clustering computed in Le Roch et al. [14]. In this paper, authors used a k-means algorithm [15, 13] to partition the genes in 15 clusters. First, we ran GOSTAT2 on each of these clusters versus the complete set of genes, and we counted the number of GO terms that are statistically over-represented with a p-value lower than 0.01. Next, in each cluster, we keep only one representative of each class of characterized homologues. The same procedure was applied to the complete data set too, and we re-ran GOSTAT2 on these modified clusters. For the *S. cerevisiae* dataset, the same entire procedure was run on a clustering of 15 classes achieved with the k-means procedure implemented in the R package[3]. Results are summarized in Table 2.

Intra-species homologies appears to highly bias the number of terms that are statistically over-represented. Actually, they are responsible for around 48% and 23% of the total number of over-represented terms in the *P. falciparum* and *S. cerevisiae* datasets, respectively. Thus, an appropriate procedure to apply the GBA principle in the non-supervised framework appears to be of great use to avoid a large proportion of false positives.

	P. falciparum				*S. cerevisiae*			
	#G	#G*	#T	#T*	#G	#G*	#T	#T*
cluster 1	70	42	18	0	565	507	58	49
cluster 2	75	68	0	0	183	132	66	66
cluster 3	85	79	0	0	287	260	65	68
cluster 4	65	50	0	0	383	348	59	52
cluster 5	64	53	0	0	48	43	0	0
cluster 6	133	126	41	29	165	143	37	25
cluster 7	75	67	17	17	313	286	16	16
cluster 8	84	77	0	0	568	478	15	3
cluster 9	64	64	0	0	183	160	25	5
cluster 10	152	133	0	0	551	481	42	37
cluster 11	78	64	15	0	121	108	48	24
cluster 12	160	144	0	0	683	570	3	1
cluster 13	114	93	0	0	591	531	86	66
cluster 14	92	77	2	0	224	196	77	66
cluster 15	72	58	10	8	688	642	36	12
total	1383	1195	103	54	5553	4885	633	490

Table 2. Bias in non-supervised methods. Columns #G and #G* denote the number of characterized genes in each cluster before and after removing the homologous genes, respectively. Columns #T and #T* denote the number of over-expressed GO term find by GOSTAT2 in each cluster before and after removing the homologous genes, respectively.

5 Bias in supervised methods

In the supervised framework, the bias occurs in the procedure used to estimate the performance of the predictor. This procedure involves randomly splitting the characterized genes into a *learning set*

[3] http://www.r-project.org/

and a *test set*. The learning set is used by the learning algorithm to build the prediction function, while the test set is used next to estimate the proportion of mistakes of the predictor when applied to new unseen genes. More precisely, the accuracy of the predictor is measured on each gene of the test set, and all the accuracy measurements are averaged to estimate a global performance. Testing the predictor on a set of genes that have not be used in the learning phase insures that the performance is not optimistically biased [12]. Moreover, a *cross-validation* (CV) procedure that averages the results of this procedure on several (*e.g.* dozens) different splits is used to provide better estimates.

However, the CV procedure described above does not take intra-species homologies into account: genes in the test set can have homologous co-expressed genes in the learning set. In this case, it is easier for the predictor to correctly annotate the genes. However, this appends more frequently for the characterized genes than for the uncharacterized ones (see Table 1). Thus, the performance estimated by CV is optimistically biased compared to the performance that can be expected on the non-characterized genes.

As in the non-supervised framework, we suppose now that uncharacterized genes with characterized homologues have been processed separately, and that we are interested in the functional annotation of all the other uncharacterized genes. How can we correct the CV in order to have an unbiased estimate of the predictor performance on these genes? This can be done in two different ways. The first one involves removing all the genes of the test set that have homologous genes in the learning set; the second one involves removing all the genes in the learning set that have homologues in the test set.

We computed the effect the bias has on a natural, GBA-based method. Of course the effect differs according to the type of predictor, but our aim here is to illustrate that it can be important. The predictor we chose is the following: Given an uncharacterized gene, it searches in the learning set the gene that has the most similar expression profile —assessed with the Pearson correlation coefficient— and gives to the uncharacterized gene the same annotations. This is a quite natural method. The performance of the predictor was assessed by CV, using the sharing score of Formula (1) to measure the accuracy between the set of predicted annotations and the real set of annotations of each tested gene. We applied a variant of the CV known as the *leave-one-out* procedure. This involves keeping only one gene for the test and using all the remaining genes to learn the prediction function. The predictor is assessed on the test gene, and the procedure is resumed until each characterized gene has been used as test. Table 3 summarizes the results achieved with and without corrections for the homology bias.

The bias also has a significant effect on the supervised methods. The impact seems to be lower than in the non-supervised framework. However, the results achieved on the *P. falciparum* dataset show that it is important to apply an appropriate CV procedure to avoid over-estimating the performance of the predictor.

6 A general scheme to properly exploit homology

As shown in the above sections, intra-species homology optimistically bias the performance of the methods used to annotate the genes without characterized homologues. On these genes, the actual performance of the methods is lower than expected. On the other hand, intra-species homologies can be exploited to obtain better predictions on the genes with characterized homologues.

In order to account for both the bias and the benefit induced by homology, a solution involves splitting the non-characterized genes into three different sets: set #1 contains the genes that possess

	no correction	correction #1	correction #2	random
P. falciparum	50.4%	43.7%	44.4%	36.8%
S. cerevisiae	43.2%	41.8%	41.3%	34.2%

Table 3. Bias in supervised methods. Accuracy estimated by a CV without correction, with the corrected CV that removes the homologues in the learning set (correction #1), and with the corrected CV that removes the homologues in the test set (correction #2). For comparison purpose, the accuracy achieved by the predictor that uses a randomly selected gene in the learning set is also reported.

characterized co-expressed homologues; set #2 contains the remaining genes that possess characterized homologues; set #3 contains all the other genes. Genes in sets #1 and #2 can then be annotated with the following supervised methods:

- Genes in set #1 are annotated with the same GO terms as their closer (assessed with the Pearson correlation coefficient) characterized co-expressed homologue; note that this is actually a form of GBA approach;
- Genes in set #2 are annotated with the same GO terms as their closer (assessed with the `blastP` E-value) characterized homologue;

Performances of these predictors are evaluated by CV on the characterized genes. This involves using the same rules to split these genese into three sets, and running two independent CVs on the first and second set. Next, genes in set #3 can be annotated using one of the corrected supervised or non-supervised GBA method described above. For example, the supervised method described in the previous section can be used.

We applied this general scheme to the *P. falciparum* and *S. cerevisiae* datasets. Results are summarized in Table 4. This table also reports the proportion of characterized and non-characterized genes that belong to each set. By extrapolating the results achieved on the characterized genes, we can hypothesize that: (1) a small proportion (1-5%) of non-characterized genes can be annotated with very high confidence ($\sim 88\%$ accuracy); (2) a larger part ($\sim$20%) can be annotated with quite good confidence (70-80% accuracy); (3) annotations of the other genes (70-80%) is more awkward. We can also compute the global performance of the method by the formula $\text{Acc} = p_1 \cdot \text{Acc}_1 + p_2 \cdot \text{Acc}_2 + p_3 \cdot \text{Acc}_3$, where p_n and Acc_n denote the proportion of uncharacterized genes and the accuracy associated with set$\#n$, respectively. We get $\text{Acc} = 53.8\%$ and 47.4% for the *P. falciparum* and *S. cerevisiae* datasets, respectively. These values are higher than the uncorrected (and optimistic) values shown in Table 3, corresponding to the standard application of GBA not accounting for homology.

	set #1			set #2			set #3		
	%C	%N	Acc	%C	%N	Acc	%C	%N	Acc
P. falciparum	24.2%	4.9%	87.5%	29.6%	22.1%	79.7%	46.2%	72.9%	43.7%
S. cerevisiae	5.1%	1.2%	88.0%	42.6%	17.4%	71.0%	52.3%	81.4%	41.8%

Table 4. Proportion of characterized (columns %C) and non-characterized (columns %N) genes in each set —for example, 24.2% of the total number of characterized genes are in set #1—, and performance of the associated predictor (columns Acc).

7 Discussion

Although this study is based on particular organisms, data, annotation systems, and GBA method, its conclusions should hold to any application of the GBA principle for functional annotation, as soon as we have more homologies within the characterized genes than between the non-characterized and the characterized genes. For example, it should apply to the GBA methods based on other information sources than transcriptomic data.

The general scheme we proposed (Section 6) could be improved in a number of ways:

- The three predictors we proposed in this scheme are over-simple and could be advantageously replaced by more sophisticated approaches. Especially, it is likely that the performance on set #3 (genes with no characterized homologue) could be easily improved.
- Co-expression and homology are defined by fixed (and relatively standard) thresholds. The co-expression threshold (0.8) was chosen to highlight the studied bias in the histograms of Figures 1 and 2. Other values could be tried and optimized to improve the accuracy of the general scheme. However, this threshold is just used to define the first two sets but does not intervene in set #3 where we use the closest neighbor for transcriptome-based predictions; thus, the gains in accuracy should not be very high.
- Selecting the homology threshold is more complex. Here again it could be appealing to try different values to improve the accuracy of the general scheme. However, this would be problematic as this threshold is the sole insurance against the bias we highlighted. The 10^{-10} value is well suited for the genomes we studied, as we observed that gene pairs with larger E-values are less likely to share common GO terms and do not show the strong bias of Figures 1 and 2. Methods should then be designed to adjust the homology threshold, aiming to increase prediction accuracy but also to prevent against bias.
- Finally, an important direction for further research would be to design methods to combine the multiple, non-homology-based information sources (transcriptomic, proteomic, interactome, etc.).

Acknowledgements

The authors thank Olivier Martin for its help and comments on this work.

Bibliography

[1] Al-Shahrour, F., Diaz-Uriarte, R., and Dopazo, J. (2004). FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics*, 20(4):578–580.

[2] Beissbarth, T. and Speed, T. (2004). GOstat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics*, 20(9):1464–1465.

[3] Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, 85:289–300.

[4] Berriz, G., King, O., Bryant, B., Sander, C., and Roth, F. (2003). Characterizing gene sets with FuncAssociate. *Bioinformatics*, 19(18):2502–2504.

[5] Boyle, E., Weng, S., Gollub, J., Jin, H., Botstein, D., Cherry, J., and Sherlock, G. (2004). GO::TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics*, 20(18):3710–3715.

[6] Brown, M. P. S., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C., Furey, T. S., Manuel Ares, J., and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data using support vector machines. *Proceedings of the National Academy of Sciences*, 1:262–267.

[7] Brun, C., Chevenet, F., Martin, D., Wojcik, J., Guenoche, A., and Jacq, B. (2003). Functional classification of proteins for the prediction of cellular function from a protein-protein interaction network. *Genome Biology*, 5(1).

[8] Castillo-Davis, C. and Hartl, D. (2003). GeneMerge–post-genomic analysis, data mining, and hypothesis testing. *Bioinformatics*, 19(7):891–892.

[9] Chen, Y. and Xu, D. (2004). Global protein function annotation through mining genome-scale data in yeast Saccharomyces cerevisiae. *Nucleic Acids Res*, 32(21):6414–6424.

[10] Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci USA*, 95(25):14863–14868.

[11] Gasch, A., Spellman, P., Kao, C., Carmel-Harel, O., Eisen, M., Storz, G., Botstein, D., and Brown, P. (2000). Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell*, 11(12):4241–4257.

[12] Hastie, T., Tibshirani, R., Botstein, D., and Brown, P. (2001). Supervised harvesting of expression trees. *Genome Biol*, 2(1).

[13] Herwig, R., Poustka, A. J., Muller, C., Bull, C., Lehrach, H., and O'Brien, J. (1999). Large-scale clustering of cDNA-fingerprinting data. *Genome Res*, 9(11):1093–105.

[14] Le Roch, K., Zhou, Y., Blair, P., Grainger, M., Moch, J., Haynes, J., Vega, P. D. L., Holder, A., Batalov, S., Carucci, D., and Winzeler, E. (2003). Discovery of gene function by expression profiling of the malaria parasite life cycle. *Science*, 301(5639):1503–1508.

[15] Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Trans. Info. Theory*, IT-2:129–137.

[16] Lockhart, D. and Winzeler, E. (2000). Genomics, gene expression and DNA arrays. *Nature*, 405(6788):827–836.

[17] Maere, S., Heymans, K., and Kuiper, M. (2005). BiNGO: a cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. *Bioinformatics*, 21(16):3448–3449.

[18] Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D., and Jacq, B. (2004). GOToolBox: functional analysis of gene datasets based on Gene Ontology. *Genome Biol*, 5(12).

[19] Mateos, A., Dopazo, J., Jansen, R., Tu, Y., Gerstein, M., and Stolovitzky, G. (2002). Systematic learning of gene functional classes from DNA array expression data by using multilayer perceptrons. *Genome Res*, 12(11):1703–1715.

[20] Shah, N. and Fedoroff, N. (2004). CLENCH: a program for calculating Cluster ENriCHment using the Gene Ontology. *Bioinformatics*, 20(7):1196–1197.

[21] Toronen, P. (2004). Selection of informative clusters from hierarchical cluster tree with gene classes. *BMC Bioinformatics*, 5.

[22] Vazquez, A., Flammini, A., Maritan, A., and Vespignani, A. (2003). Global protein function prediction from protein-protein interaction networks. *Nat Biotechnol*, 21(6):697–700.

[23] Wu, L., Hughes, T., Davierwala, A., Robinson, M., Stoughton, R., and Altschuler, S. (2002). Large-scale prediction of saccharomyces cerevisiae gene function using overlapping transcriptional clusters. *Nat Genet*, 31(3):255–265.

[24] Zhou, Y., Young, J., Santrosyan, A., Chen, K., Yan, S., Winzeler, E., Young, J., Fivelman, Q., Blair, P., de la Vega P, KG, L. R., Zhou, Y., Carucci, D., Baker, D., and Winzeler, E. (2005). In silico gene function prediction using ontology-based pattern identification. *Bioinformatics*, 21(7):1237–1245.