Metric learning on expression data for gene function prediction

Makrodimitris, Stavros; Reinders, Marcel J.T.; van Ham, Roeland C.H.J.

DOI
10.1093/bioinformatics/btz731

Publication date
2020

Document Version
Final published version

Published in
Bioinformatics (Oxford, England)

Bioinformatics, 36(4), 2020, 1182–1190
doi: 10.1093/bioinformatics/btz731
Advance Access Publication Date: 28 September 2019
Original Paper

Gene expression

Metric learning on expression data for gene function prediction

Stavros Makrodimitris [1,2,*], Marcel J. T. Reinders[1,3] and Roeland C. H. J. van Ham[1,2]

[1]Delft Bioinformatics Lab, Delft University of Technology, Delft 2628 XE, The Netherlands, [2]Keygene N.V., Wageningen 6708 PW, The Netherlands and [3]Leiden Computational Biology Center, Leiden University Medical Center, Leiden 2333 ZC, The Netherlands

*To whom correspondence should be addressed.

Associate Editor: Mark Robinson

Received on June 4, 2019; revised on August 31, 2019; editorial decision on September 13, 2019; accepted on September 25, 2019

Abstract

Motivation: Co-expression of two genes across different conditions is indicative of their involvement in the same biological process. However, when using RNA-Seq datasets with many experimental conditions from diverse sources, only a subset of the experimental conditions is expected to be relevant for finding genes related to a particular Gene Ontology (GO) term. Therefore, we hypothesize that when the purpose is to find similarly functioning genes, the co-expression of genes should not be determined on all samples but only on those samples informative for the GO term of interest.

Results: To address this, we developed Metric Learning for Co-expression (*MLC*), a fast algorithm that assigns a GO-term-specific weight to each expression sample. The goal is to obtain a weighted co-expression measure that is more suitable than the unweighted Pearson correlation for applying Guilt-By-Association-based function predictions. More specifically, if two genes are annotated with a given GO term, *MLC* tries to maximize their weighted co-expression and, in addition, if one of them is not annotated with that term, the weighted co-expression is minimized. Our experiments on publicly available *Arabidopsis thaliana* RNA-Seq data demonstrate that *MLC* outperforms standard Pearson correlation in term-centric performance. Moreover, our method is particularly good at more specific terms, which are the most interesting. Finally, by observing the sample weights for a particular GO term, one can identify which experiments are important for learning that term and potentially identify novel conditions that are relevant, as demonstrated by experiments in both *A. thaliana* and *Pseudomonas Aeruginosa*.

Availability and implementation: *MLC* is available as a Python package at www.github.com/stamakro/MLC.

Contact: s.makrodimitris@tudelft.nl

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Knowing which biological processes and pathways are affected by each gene would be a useful tool for plant biologists and breeders. With this information, they can more easily identify genes that are likely to affect the phenomenon or trait they are studying and prioritize genes for experimental testing. The Biological Process Ontology (BPO) of the Gene Ontology (GO) (Ashburner *et al.*, 2000) provides us with a set of terms that describe biological processes at different levels of granularity and can be used to annotate genes from all species in a systematic way. However, the use of computational methods to accurately predict BPO annotations, also known as Automatic Function Prediction (AFP), remains challenging, as demonstrated in the Critical Assessment of Functional Annotation (CAFA) challenges (Jiang *et al.*, 2016a).

Most AFP methods use the Guilt-By-Association (GBA) principle. They define a similarity or dissimilarity measure between genes and use it as a proxy for functional similarity. Then, they assign GO annotations to genes of unknown function based on the functions of the genes most similar to them. The choice of similarity measure is always motivated by biology. For instance, sequence similarity points towards a conserved structure which in turn implies similar function. Alternatively, co-expression across different conditions may hint at involvement in the same pathways. Recent results from the third CAFA challenge hint at the great potential of gene expression data to find which genes are involved in a specific biological process (Zhou *et al.*, 2019). Combining multiple similarity measures in order to better approximate functional similarity is also possible, as done for instance in (Cozzetto *et al.*, 2013; Lan *et al.*, 2013; Zhang *et al.*, 2017).

Genes that are involved in the same biological processes are expected to show similar expression patterns, as they respond similarly to perturbations related to these processes. Discovering BPO annotations for all unannotated genes requires data from a wide

range of different experimental conditions. For example, we need samples from different tissues, different time points across development, from wild-type or mutant plants etc. Thanks to world-wide sequencing efforts, more and more RNA-Seq data are becoming available to public databases, such as ArrayExpress (Parkinson *et al.*, 2007) and GEO (Clough and Barrett, 2016).

The Pearson Correlation Co-efficient (*PCC*) is the most widely used measure of gene co-expression similarity and has been largely successful, especially for microarray-derived expression data. For instance, for MS-kNN (Lan *et al.*, 2013), one of the top-performing methods in CAFA2 (Jiang *et al.*, 2016a), the *PCC* was calculated on samples from 392 human microarray datasets to quantify co-expression similarity between genes, outperforming sequence similarity for AFP in BPO (Lan *et al.*, 2013).

PCC might, however, not be the optimal co-expression measure due to the diversity of biological processes and heterogeneity of public expression datasets. This means that only a subset of all available experimental conditions is likely to be truly informative about a specific GO term. For example, let us assume that we are looking for genes involved in plant immune response. Using the *PCC* across all possible conditions, we implicitly expect that all such genes are expressed similarly not only during immune response, but across all conditions and tissues. However, differential co-expression analysis has shown that several *Arabidopsis thaliana* immune genes, such as FLS2, ADR1 and JAR1, change co-expressed partners before and after infection with *Pseudomonas syringae* (Jiang *et al.*, 2016b). A gene that is co-expressed with immune genes during (only) infection is still a good candidate gene for immune response, even if it has different expression patterns to the immune genes in other tissues or developmental stages. Including many unrelated expression samples, essentially adds noise to the correlations. According to this reasoning, we should be able to improve the performance of co-expression-based gene function prediction by calculating co-expression only over the samples that are relevant for each term.

This insight that the *PCC* might be suboptimal is not new. For example, Jaskowiak *et al.* showed that k-means clustering of gene expression data heavily relies on the choice of similarity measure (*PCC*, Spearman correlation, Euclidean distance etc.) and that the most suitable measure varies across different datasets (Jaskowiak *et al.*, 2012). As another example, Hu *et al.* showed that using an inappropriate distance metric can really harm the performance of the *k*-Nearest Neighbors (*k*-NN) classifier in biomedical datasets (Hu *et al.*, 2016).

Adapting a distance measure is a subfield within machine learning that is called metric learning: learning a distance function from a dataset of examples that can most effectively be utilized to perform a task, e.g. discriminating between two classes. It is most explored in combination with the *k*-NN classifier (Bellet *et al.*, 2013). In the context of AFP, Ray and Misra developed a metric learning method called Genetic Algorithm for Assigning Weights to Gene Expressions using Functional Annotations (*GAAWGEFA*) that learns a weighted *PCC* on microarray data using a genetic algorithm to find the optimal values for the weights (Ray and Misra, 2019). They showed that their weighted correlation increases the protein-centric precision compared to *PCC* in a yeast dataset. Metric learning has also been applied to AFP combined with multiple-instance learning (Xu *et al.*, 2017). In that work, each protein is viewed as a 'bag of domains' and metric learning is used to learn a distance function between proteins (based on their domains) that is representative of functional similarity.

Here, we use metric learning to identify the most informative conditions for a given GO term. Similar to *GAAWGEFA*, our goal is to assign a weight to every RNA-Seq sample. *GAAWGEFA* learns one weighted *PCC* for all GO terms (Ray and Misra, 2019). On the contrary, our approach, Metric Learning for Co-expression (*MLC*), optimizes the weights per term. Our philosophy (graphically shown in Fig. 1) is that weights should be chosen in such a way that a pair of genes annotated with the same term should have maximally similar expression profiles, i.e. comply with our assumption that these genes should be co-expressed. On the other hand, when one gene is annotated with the term and the other not, we expect that such a pair should not have high co-expression. In other words, we would like to select weights that minimize the co-expression for these pairs. For pairs of genes both not annotated with the term, we cannot say anything about the co-expression since they might be annotated with another term, and thus be co-expressed too (albeit for other conditions/samples). Consequently, the co-expression of these pairs should be ignored when optimizing weights for the GO term of consideration. A high weight for a sample will put emphasis on that sample when calculating the co-expression over all samples, whereas a low weight for a sample will reduce the influence of that sample. When a weight becomes zero the sample is even ignored. To enforce selecting informative samples, we additionally apply an L1 sparsity constraint on the weights, which will set a weight to zero when a sample is uninformative (Tibshirani, 1996). In contrast to *GAAWGEFA*, where they have used a genetic algorithm to find the weights, we are able to pose the weight optimization in an elegant mathematical formulation that can be minimized efficiently using standard methods. To reduce the computational burden even further, we use the weighted inner product as a similarity function instead of the weighted *PCC*. We evaluate our algorithm on public RNA-Seq data from *A. thaliana* and on the microarray data from *Pseudomonas aeruginosa* that were used for the CAFA-π challenge, which included experimental validation of gene functions. (Zhou *et al.*, 2019).

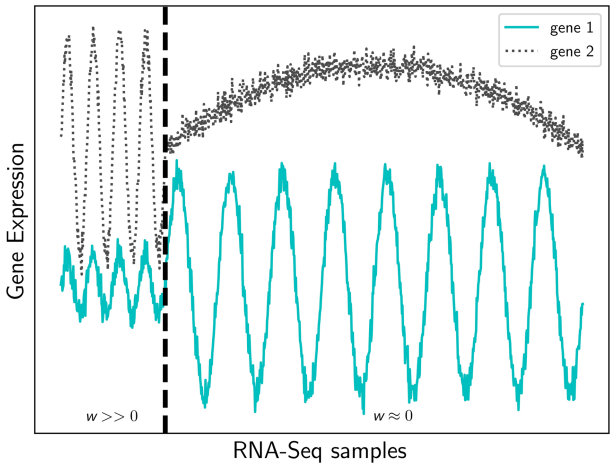

Fig. 1. Illustrative example of the expression of two hypothetical genes (*y*-axis, solid and dashed lines) involved in the same biological process over a large set of samples (*x*-axis). The total Pearson correlation between the genes is 0.09. *MLC* sets large weight values (w_m) for the samples left of the vertical dashed line (where the unweighted correlation is 0.92) and small or zero weights for the samples on the right (unweighted correlation = 0.002)

2 Materials and methods

2.1 Data and preprocessing

We used the API of the European Bioinformatics Institute (Petryszak *et al.*, 2017) to download all *A. thaliana* RNA-seq studies available at ArrayExpress (Parkinson *et al.*, 2007). All samples had been processed using the same pipeline and expression was measured using raw read counts. We restricted our dataset to samples that used the latest version of the *A. thaliana* genome (TAIR10) and had fewer than 10% unmapped reads. After removing duplicate experiments, we had 4215 samples from 298 different studies (batches) for 32 833 genes, 26 925 of which were protein-coding. We used a preprocessing pipeline similar to the one used to construct the ATTED-II RNA-Seq co-expression network (Obayashi *et al.*, 2018). We first removed samples with fewer than 10 000 000 mapped reads. Then, we removed lowly expressed genes (genes with maximum expression over the remaining samples less than 100). To diminish the zero-inflation of the dataset, we also removed all genes that were not

expressed in at least half of the samples (median expression smaller than 1). Finally, we log-transformed the expression counts using a pseudocount of 0.125. Supplementary Material S1 and Supplementary Figure S1 describe the results when excluding the filtering of low-coverage samples.

We then mapped the TAIR gene ID's to UniProt ID's. BPO annotations were downloaded from GOA (https://www.ebi.ac.uk/GOA) in September 2016 and annotations with the IEA evidence code were removed. A total of 2978 samples and 6013 genes with BPO annotations remained after these filtering steps. We applied ComBat (Johnson *et al.*, 2007) to remove unwanted variation stemming from the fact that the different samples come from different studies (batch effects). ComBat uses a Bayesian method to standardize the mean and the variance of each gene in each study (batch). In order to be able to estimate within-batch variances, we removed all studies that had only one sample, leaving us with 2959 samples (Supplementary Material S2).

For the experiments on *P. aeruginosa* we used a pre-processed microarray expression compendium from (Tan *et al.*, 2017). This dataset contains the expression of 5549 genes measured over 1051 samples. The gene annotations were downloaded from the Supplementary Material of the CAFA-π paper (Zhou *et al.*, 2019).

2.2 Notation

We use $\mathbf{x}_i \in \mathbb{R}^f$ to denote the expression of gene i across all $f = 2959$ samples. x_{im} is the expression of gene i at sample m. $\bar{x}_i$ is the mean of gene i across all samples. Given N genes and a GO term l, we denote as $\mathbf{y}(l) \in \{0,1\}^N$ the vector with the class labels of the genes, with $y(l)_i = 1$ iff gene i is annotated with l. The sample weights are represented by a vector with f non-negative elements $\mathbf{w}(l) \in [0, +\infty)^f$.

2.3 Weighted and unweighted measures of Co-Expression

The most widely-used measure of co-expression between two gene expression vectors $\mathbf{x}_i, \mathbf{x}_j$ is the Pearson Correlation Coefficient (*PCC*), which is defined as follows:

$$PCC(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{m=1}^{f}(x_{im} - \bar{x}_i)(x_{jm} - \bar{x}_j)}{\sqrt{\sum_{m=1}^{f}(x_{im} - \bar{x}_i)^2}\sqrt{\sum_{m=1}^{f}(x_{jm} - \bar{x}_j)^2}} \quad (1)$$

Note that the numerator is the covariance between $\mathbf{x}_i$ and $\mathbf{x}_j$ and the denominator is the product of the SDs of the two vectors.

A related, but simpler measure is the inner product similarity (*S*), which, on the contrary, is sensitive to the mean expression of both genes:

$$S(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \cdot \mathbf{x}_j = \sum_{m=1}^{f} x_{im}x_{jm} \quad (2)$$

If two vectors $\mathbf{x}_i$, $\mathbf{x}_j$ both have zero mean and unit L2-norm, then their *PCC* is equal to their inner product. This equality does not hold anymore if we weigh each vector element (sample) differently. However, since the two metrics are related, we chose to use the weighted inner product similarity instead of the weighted *PCC* as our expression similarity function in order to simplify the problem. We center and scale our data so that the (unweighted) mean of every gene is zero across all conditions and its (unweighted) L2-norm is equal to one:

$$\tilde{\mathbf{x}}_i = \frac{\mathbf{x}_i - \bar{x}_i}{\|\mathbf{x}_i - \bar{x}_i\|} \quad (3)$$

Then, we define our similarity function as the weighted inner product of the two scaled expression vectors ($S_\mathbf{w}$):

$$S_\mathbf{w}(\mathbf{x}_i, \mathbf{x}_j) = \tilde{\mathbf{x}}_i^T \cdot W \cdot \tilde{\mathbf{x}}_j = \sum_{m=1}^{fw_m} \tilde{x}_{im}\tilde{x}_{jm} \quad (4)$$

Where $W = \mathrm{diag}(\mathbf{w})$ is a diagonal matrix containing the sample weights.

2.4 *MLC*

The rationale for learning the weights is to maximize the performance of the *k*-NN classifier. For this purpose, we want the expression similarity between two genes that are both annotated with a given GO term l to be higher (on average) than the similarity between a gene that is annotated with l and a gene that is not. We group each gene pair into one of the following three categories:

i. Both genes are annotated with l (we call these 'positive–positive pairs' or '*p–p*').
ii. Exactly one of the two genes is annotated with l ('positive–negative pairs' or '*p–n*').
iii. Neither gene is annotated with l ('negative–negative pairs' or '*n–n*').

Our goal is to find the weight values w_m that maximize the separability between '*p–p*' and '*p–n*' pairs. Let μ_{p-p}, σ^2_{p-p} denote the mean and variance of the weighted similarity value S_w of all '*p–p*' gene pairs and, similarly, μ_{p-n}, σ^2_{p-n} for all '*p–n*' gene pairs. Let also N_{p-p} and N_{p-n} denote the number of gene pairs in each category. We use Welch's two-sample *t*-statistic with unequal variances to quantify the notion of separability:

$$t(\mathbf{w}) = \frac{\mu_{p-p} - \mu_{p-n}}{\sqrt{\frac{\sigma^2_{p-p}}{N_{p-p}} + \frac{\sigma^2_{p-n}}{N_{p-n}}}} \quad (5)$$

Note that μ and σ^2 are functions of $\mathbf{w}$, but this dependence is not shown explicitly in Equation (5) to keep the notation simple. Maximizing $t(\mathbf{w})$ is equivalent to minimizing $-t(\mathbf{w})$. In order to enable sample selection, we also added an L1 regularization term that forces the weights of uninformative samples to zero (Tibshirani, 1996). Our optimization problem then becomes:

$$\min_{\mathbf{w}} \left[-\alpha t(\mathbf{w}) + (1-\alpha)\sum_{m=1}^{f} w_m \right], \quad s.t. \quad w_m \geq 0 \forall m \quad (6)$$

Parameter α controls the trade-off between the actual cost and the regularization. The minimization of Equation (6) is done with the Broyden-Fletcher-Goldfarb-Shanno method (Byrd *et al.*, 1995).

2.4.1 Global *MLC*

To investigate the effect of creating GO-term specific predictors, we also implemented a version of *MLC* that is applicable to all terms simultaneously. To this purpose, we redefined '*p–p*' gene pairs as pairs of two genes which share at least one GO term and '*p–n*' pairs as pairs of two genes that share no GO annotations. All the ensuing steps remain the same as for the term-specific *MLC*. We call this method 'Global *MLC*' (MLC_G).

2.5 Experimental set-up

We compared *MLC* to the unweighted *PCC* baseline. To investigate the effect of the use of a term-specific classifier, we created term-specific classifiers from the *PCC* by tuning the classifier parameter *k* individually per GO term and not globally over all terms. We called this approach *PCC(k)*. We also compared to *GAAWGEFA* which, like *MLC*, learns a weighted co-expression measure (Ray and Misra, 2019). *GAAWGEFA* is not GO-term-specific and optimizes the mean protein-centric precision using a genetic algorithm, so we also constructed a non-term-specific version of *MLC* (MLC_G) to compare against. Another way to measure co-expression is the Mutual Rank (*MR*) (Obayashi *et al.*, 2018) which is used in the ATTED-II database. Although *MR* neither selects samples nor weighs samples differently, it has been shown to outperform the *PCC* for function prediction (Obayashi *et al.*, 2018), so we included it in the comparison as a stronger baseline. Input to *MR* are typically the *PCC* values of all gene pairs, although it can be applied to any co-expression measure. More details on the definition and implementation of each of these methods are given in Supplementary Material S3.

We evaluated the methods in three ways: (1) A cross-validation experiment using all *A. thaliana* genes with at least one GO annotation (reported as 'CV results'), (2) a time-course experiment using the preliminary test set from CAFA3 containing 6077 training and 90 test genes also from *A. thaliana* (reported as 'CAFA3 results') and (3) the dataset of CAFA-π from bacterium *P. aeruginosa*, where the goal was to make term-specific predictions for biofilm formation and motility (Zhou *et al.*, 2019). For the cross validation experiment, we used nested cross-validation (Varma and Simon, 2006), using the inner loop to optimize the parameters and the outer loop to evaluate performance on previously unseen genes (Supplementary Fig. S2, Supplementary Material S4). As in this work we are dealing with the problem of identifying which genes should be annotated with a specific GO term, we focus on term-centric evaluation using the mean *ROCAUC*. Details about the three evaluation modes are provided in Supplementary Material S4 and details about the used term-centric evaluation metrics as well as protein-centric metrics that we also used are given in Supplementary Material S5.

3 Results

3.1 All methods outperform the *PCC*

We compared our metric learning approach (*MLC*), as well as *MR* and *GAAWGEFA* to the standard, unweighted *PCC* using 3-fold cross-validation. *PCC* achieved a mean term-centric *ROCAUC* of 0.69, while the performance of both *MR* and *MLC* with the weighted inner product was 0.72 (Table 1). The performance of *GAAWGEFA* was 0.71. MLC_G, the non-term-specific version of *MLC*, also achieved a mean *ROCAUC* of 0.72. Although all methods perform fairly similarly according to protein-centric measures (Supplementary Tables S1–2, Supplementary Material S6), *PCC* performs significantly worse than the other methods on term-centric *ROCAUC* [False Discovery Rate (FDR) < 0.036, Table S3–6, Supplementary Material S7, effect size 4%]. This shows that the *PCC* is indeed a suboptimal co-expression measure. When randomly permuting the GO annotations of the genes, both *PCC* and *MLC* had a mean *ROCAUC* of 0.5, i.e. equal to random guessing, implying that *MLC* does not artificially generate information in the absence of real structure (Supplementary Material S8).

3.2 *MLC* is the best at predicting specific GO terms

Although *MR*, *GAAWGEFA* and *MLC* perform equally on average, one is typically not interested in predicting GO terms that are 'near' the ontology root, as most of them describe too general biological processes (Clark and Radivojac, 2013). Therefore, we compared the performances of these methods as a function of term specificity. As measures of specificity, we used the maximum path length to the ontology root and the Resnik Information Content (IC) (Resnik, 1995). One way to take term specificity into account is to calculate the weighted term-centric *ROCAUC*, where each term is weighted by its IC when calculating the average. As shown in Table 1, *MLC* achieves the highest weighted *ROCAUC*. The difference is statistically significant for all methods except for *MR* (Supplementary Table S4, Supplementary Material S7), although the effect size is small (1.5%).

Furthermore, we grouped the GO terms into quintiles (quantiles at 0, 20%, 40%, 60% and 80%) and plotted the distribution of the percent differences in performance of *MLC* from *MR* for each quintile (Fig. 2a). We observed that for the first two quintiles (i.e. the 40% most frequent terms), *MLC* performs worse than *MR*, while for the 60% most specific terms, both the mean and the median performance of *MLC* is better. Further analysis showed that for the very general terms, *MLC* makes a lot more type I errors (false positives) than for the more specific ones (Supplementary Fig. S3, Supplementary Material S9) and that makes it underperform with respect to *MR*. The Spearman correlation between percent difference and Resnik IC was 0.26. The same pattern is evident when comparing *MLC* to all other methods (*PCC*, *GAAWGEFA* and MLC_G), as well as when replacing Resnik IC with the path length to the ontology root (Supplementary Tables S7 and 8, Supplementary Figs. S4 and 5, Supplementary Material S10). From that we can conclude that term-specific *MLC* is the preferred method for finding genes belonging to rarer terms.

3.3 *MLC* tunes the weights to find '*p*–*p*' pairs

The goal of *MLC* is to choose the weights so that for test genes that have a particular GO annotation, the learned similarities are higher to training genes that have the same annotation than to genes that do not. As an example, Figure 2b and c show the distribution of co-expression similarities between the test genes annotated with term GO: 1903047 (mitotic cell cycle process) and all training genes for the *PCC* and *MLC* similarities, respectively. It is clear that for *MLC*, the test genes are a lot more similar to training genes with the same annotation. Note, however, that, for this term, a significant portion of the similarities are negative (small blue peak in Fig. 2b). This means that some positive genes are anti-correlated to the rest. For these cases *MLC* will make Type II errors (false negatives).

Figure 2d shows the distribution of the number of selected samples for each GO term. For about 33% of all GO terms, *MLC* selected less than 9% of the available samples (252 or fewer), setting all other weights to zero. The median number of selected samples was 2035 out of 2959 or about 69% of all samples. Randomly selecting samples did not improve the mean performance of the *PCC* (Supplementary Material S11), implying that the correct samples have to be selected for each term. Moreover, for about 23% of the terms, *MLC* kept all the samples and weighted them more or less equally (maximum SD of weights = 0.006), in which case *MLC* was almost equivalent to the unweighted inner product. As expected, for those terms *MLC* had on average similar performance to the baseline *PCC*. We also found that individually tuning k per GO term for the *PCC* gave on average the same term-centric *ROCAUC* as the baseline *PCC* [*PCC (k)*, Table 1], so the performance improvement is not caused by simply choosing the optimal k value for each GO term. Finally, we observed a small negative correlation between term IC and the number of samples selected (Spearman $\rho = -0.09$, P-value $= 0.057$). This means that *MLC* has a slight tendency to

Table 1. Mean term-centric *ROCAUC* achieved by the methods under comparison using 3-fold cross-validation (CV, second and third column) and when testing on the CAFA3 dataset (CAFA3, fourth and fifth column)

Method	*ROCAUC* (CV)	Weighted *ROC AUC* (CV)	*ROC AUC* (CAFA3)	Weighted *ROCAUC* (CAFA3)
PCC	0.69 ± 0.003	0.69 ± 0.003	0.68 [0.63, 0.72]	0.68 [0.63, 0.73]
PCC(k)	0.69 ± 0.003	0.69 ± 0.003	0.68 [0.63, 0.71]	0.68 [0.63, 0.72]
PCC + MR	0.72 ± 0.002	0.72 ± 0.002	0.69 [0.65, 0.73]	0.69 [0.65, 0.73]
GAAWGEFA	0.71 ± 0.002	0.71 ± 0.002	0.69 [0.65, 0.73]	0.70 [0.65, 0.74]
MLC (Sw)	0.72 ± 0.003	**0.73 ± 0.003**	0.69 [0.65, 0.73]	0.69 [0.65, 0.73]
MLC_G	0.72 ± 0.003	0.72 ± 0.003	**0.71 [0.67, 0.75]**	**0.72 [0.67, 0.76]**
MLC-MR Hybrid	**0.73 ± 0.005**	**0.73 ± 0.005**	0.69 [0.65, 0.73]	0.69 [0.66, 0.73]

Notes: For the cross-validation, we report the average performance over the three folds as well as the corresponding standard error. For the CAFA3 results we report the performance on the test set as well as the 95% confidence intervals from doing 1000 bootstrapped tests.
The top performance of every column is shown in bold.

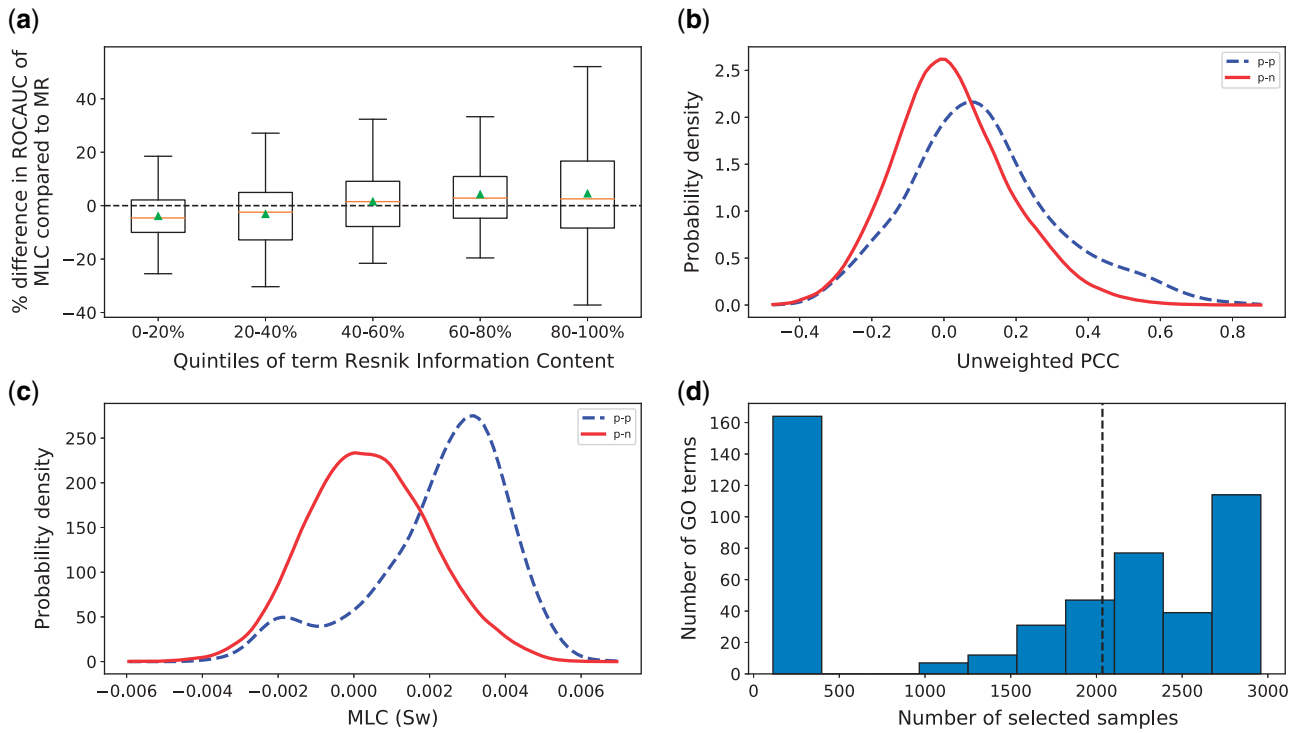

Fig. 2. (**a**) Percent increase in *ROCAUC* of *MLC* (S_w) with respect to *MR* as a function of Resnik IC. For each set of terms in each quintile of IC, the corresponding box includes the two middle quartiles of the percent increase for these terms. An orange line denotes the median and a green triangle the mean. The error bars extend to 1.5 times the range of the two middle quartiles. Note, that the 0–20% quintile corresponds to the 20% least specific terms and the 80–100% quintile to the 20% most specific ones. (**b–c**) Distributions of co-expressions for genes annotated with term GO: 1903047. In dashed blue lines, the co-expression values between a test and a training gene that both are annotated with that term. In solid red lines, the co-expression between test genes annotated with that GO term and training genes that are not. Co-expression is measured as the *PCC* (b) and the S_w trained by *MLC* (c). The *x*-axis shows the co-expression values and the *y*-axis the probability density estimated with Gaussian kernels. Note that the *PCC* and S_w have different ranges due to the weight optimization. (**d**) Histogram of the number of samples that were selected for each GO term. The *x*-axis corresponds to the number of selected samples and the *y*-axis to how many GO-term-specific similarity functions selected that many samples. The dashed line denotes the median number of non-zero weights

select fewer samples for more specific terms, but this result is not statistically significant.

3.4 The weights learned by *MLC* help at identifying relevant experimental conditions

The weights learned by *GAAWGEFA* are roughly uniformly distributed between 0 and 1 (Kolmogorov–Smirnov test statistic = 0.011, *P*-value = 0.847) and are not correlated to any of the term-specific weight profiles of *MLC*, which tend to have an exponential-like distribution (Supplementary Figs. S6 and 7, Supplementary Material S12), as many samples get a weight of zero. Furthermore, samples from the same study (batch) tend to be either selected or not selected together by *MLC* (Supplementary Table S9, Supplementary Material S13).

We used simulated data to validate the sample selection of *MLC*. We created an artificial dataset with 7000 genes, 3000 samples and three GO terms, where the genes with a particular GO term are correlated over a predefined set of samples. The sets of informative samples are of equal length and do not overlap (Supplementary Fig. S8, Supplementary Material S14). We varied the number of informative samples from 10 to 500 and compared *MLC* to the *PCC* calculated over all samples and the *PCC* calculated only over the informative ones [which can be seen as the optimal, ground-truth (GT) performance]. As seen in Figure 3, when up to 200 samples drive the similarity, *MLC* performs considerably better than the *PCC* and only slightly worse than the GT *PCC* (*GT*). When the number of informative samples increases, both *MLC* and *PCC* increase performance, eventually both converging to *GT*. Yet, *MLC* converges much faster, thus achieving *GT PCC* performance with fewer informative samples. We also found that in all cases, and for all numbers of relevant samples except 500, the *MLC* weights of the GT samples were on average significantly larger than those in the rest of the samples (two-sample *t*-test, FDR ¡ 0.01, Supplementary Fig. S9a, Supplementary Material S14). Moreover, there was a significant enrichment of the informative samples in the samples selected by *MLC* (Fisher's exact test, FDR ¡ 0.01, Supplementary Fig. S9b, Supplementary Material S14). When the number of informative samples increased to 500, *MLC* selected all the samples with rather similar weights (Fig. S9) and performed consistently equal to both *PCC* and *GT*.

Continuing on the example from before, for term GO: 1903047 (mitotic cell cycle process) *MLC* gives the highest weight to a sample of a plant grown in the absence of phosphorus, which has been shown to restrict the cell division rate (Kavanová *et al.*, 2006). Among the samples with highest weights are also many samples from experiments studying seed germination, a process closely linked to cell cycle (Vázquez-Ramos and de la Paz Sánchez, 2003). Finally, two *IBM1* mutant samples were selected with very high weights. The *IBM1* gene codes for histone demethylation protein and has GO annotations that include flower, root and pollen development. The complete weight profile is shown in Supplementary Figure S10 (Supplementary Material S15).

As another example, we looked at 'regulation of flower development' (GO: 0009909). For this term, *MLC* selected 164 samples and achieved ROC AUC of 0.74 while the *PCC* score was 0.55. The top scoring sample came from an AS1 mutant plant. AS1 is a well-known transcription factor, key in many developmental processes including flowering (Xu *et al.*, 2008). Several wild-type samples were selected, many of them from meristems, tissues which contain undifferentiated cells and drive tissue differentiation. In the top-10 samples, we also found a CORYNE mutant [CORYNE is involved in the signaling of cell differentiation in flowers (Muller *et al.*, 2008)] and an AGO1 mutant, a gene crucial for miRNA-based mRNA splicing (Vaucheret *et al.*, 2004). Finally, 3 out of the top-10 samples were from knock-outs of well-known DNA methylation

genes. Chen *et al.* were—to our knowledge—the first to extensively study the interplay between epigenetics, transcription factors and miRNAs in flower development, which is otherwise largely not understood (Chen *et al.*, 2018). *MLC* was able to highlight that all these three regulatory mechanisms are at the same time informative for understanding flower development, without using the data from that work or any Chip-Seq or miRNA data.

Together these examples highlight that *MLC* has the ability to identify novel experimental conditions for a specific GO term. Moreover, the weights learned by *MLC* are also consistent with the ontology structure and the existing annotations, as shown in Supplementary Figure S11 in Supplementary Material S16.

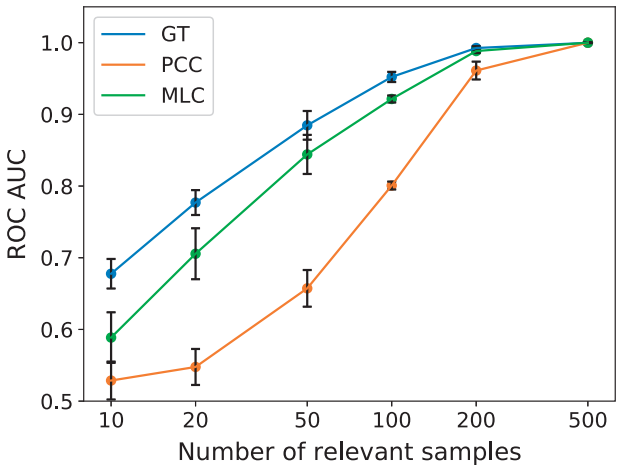

Fig. 3. *ROCAUC* (*y* axis) on simulated data as a function of the number of relevant samples (*x* axis in log scale) for *MLC* (green), *PCC* (orange) and the *PCC* using only the GT samples (blue). The error bars denote the SD over five repetitions

3.5 Using all samples obscures co-expression

Next, we investigated the terms for which *MLC* performed sample selection, i.e. assigning a non-zero weight to at most 9% of the samples. We looked at the *PCC* values for '*p–p*' and '*p–n*' gene pairs. Figure 4a shows an example of the distributions of the *PCC* values for '*p–p*' and '*p–n*' pairs for term 'GO: 1903047' (mitotic cell cycle process). Next, we calculated the *PCC* for all '*p–p*' and '*p–n*' pairs, but only considering the samples that were selected by *MLC* (i.e. had a non-zero weight assigned), as well as only considering the samples that were not selected by *MLC* (i.e. were assigned a weight of zero), shown in Figure 4b and c, respectively. One can notice that the two distributions ('*p–p*' and '*p–n*' *PCC* values) differ more when taking the *MLC* selected samples in consideration (compare Fig. 4a with Fig. 4b). When only considering the samples that were not selected, the two distributions differ in a similar way to when all samples are being considered (compare Fig. 4a with Fig. 4c).

For every GO term for which *MLC* performs sample selection, we calculated the mean *PCC* of all '*p–p*' and '*p–n*' pairs under the three sample sets (all samples, not selected samples and selected samples). Figure 4d shows the average of these values of all GO terms. We also performed 1000 bootstraps, sampling GO terms with replacement to obtain 95% confidence intervals for these averages. We observed that the difference in mean co-expression between '*p–p*' and '*p–n*' in the samples not selected by *MLC* is similar to the difference in all the samples. Although these differences are statistically significant, they are also significantly smaller than the difference in the *MLC*-selected samples (bootstrap *P*-value ¡0.001) (Fig. 4d). This means that although the whole dataset does contain a few samples that are informative for these GO terms, calculating the co-expression over a larger set of samples can corrupt the 'real' co-expression signal, increasing the difficulty of discovering new genes that play a role in these processes.

3.6 Combining *MR* and *MLC*

The performances of *MLC* and *MR* are positively correlated (Spearman $\rho = 0.13$, *P*-value =0.003). We also applied *MR* on the co-expression similarities obtained with *MLC*, as *MR* is in principle

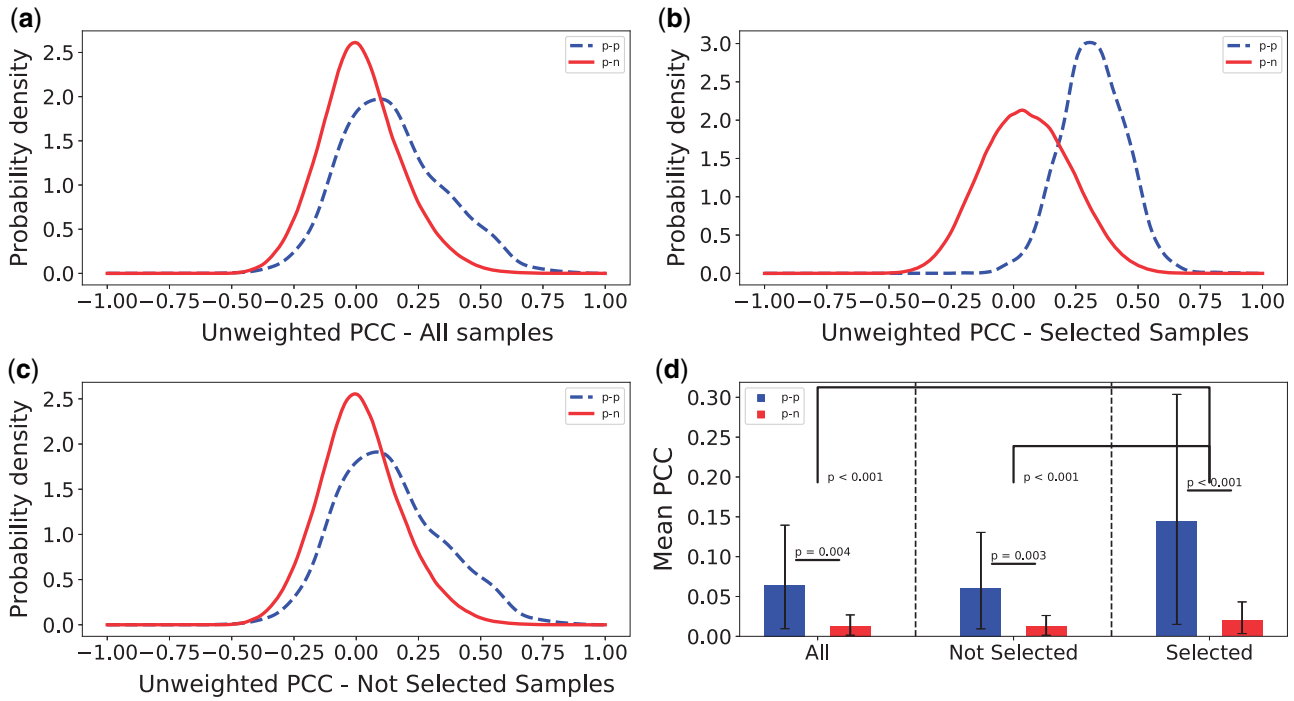

Fig. 4. (**a–c**) Distributions of Pearson correlations for pairs of training genes that are both annotated with term GO: 1903047 ('*p–p*', blue dashed), and for pairs of training genes of which only one is annotated with that term ('*p–n*', red solid). The correlations are calculated using all samples (a), the samples that were selected by *MLC* (b) and the samples that were not selected (c). (**d**) The means of the distributions of (a–c), over all GO terms where less than 10% of the samples where used to calculate the co-expression (more than 90% zero weights). Values for '*p–p*' pairs are colored blue and for '*p–n*' pairs red. The error bars show the 95% confidence intervals for the means, calculated with 1000 bootstraps. Above the bars the bootstrap *P*-values are shown for the pairwise comparisons

not restricted to using only the *PCC*. We found a small improvement compared to standalone *MLC*, with a mean ROC AUC of 0.73. Also, the performances of *MLC* and *MLC* + *MR* were highly correlated (Spearman $\rho = 0.97$, *P*-value $\ll 10^{-20}$).

We tried another approach to combine *MLC* and *MR* depending on the performance of the methods. If the training *ROCAUC* of *MR* was larger than 0.8 for a GO term, we used the predictions of *MR* for that term, otherwise we used the predictions of *MLC*. This combined classifier had an incrementally larger term-centric *ROCAUC* (0.73, Table 1–Hybrid), though statistically significant (*P*-value = 0.008, two-sample *t*-test). The threshold of 0.8 training *ROCAUC* was chosen arbitrarily and was not tuned to maximize performance. This naïve hybrid classifier shows that there is potential to improve performance by combining *MLC* and *MR* in more sophisticated ways.

3.7 CAFA results

Moreover, we benchmarked *MLC* on 90 temporary *A. thaliana* targets from the CAFA3 competition. The results are similar, both in absolute numbers and the ranking of the methods (Table 1), with both *MR* and *MLC* outperforming the *PCC* on average. However, due to the small size of the dataset, the confidence intervals are much wider, so no significant conclusions can be drawn.

Lastly, we compared *MLC* and *MR* to *PCC* in the term-centric challenge CAFA-π for *P. aeruginosa*, where *PCC* on microarray data was the top-performing method (Zhou *et al.*, 2019). The goal of the challenge was to predict proteins involved in motility and biofilm formation and the ground truth was obtained using genome-wide assays. Note that these two terms are relatively frequent (about 14% and 12% of the tested genes were annotated with biofilm and motility, respectively). Comparing them to our experiments in Arabidopsis (Fig. 2a), they would fall in the left-most, least informative bin, where *MLC* is expected to be on average slightly worse than *MR* and about similar to *PCC*. The results show that *PCC*, *MR* and *MLC* achieve similar *ROCAUC* (around 0.6 for both biofilm and motility, which is also very similar to the performance reported in Zhou *et al.*, 2019) (Supplementary Table S10, Supplementary Material S17). However, *MLC* selected samples from relevant conditions (Supplementary Material S17), enabling interpretability of the predictions. We also examined the genes for which *MLC* was most confident for their involvement in either biofilm formation or motility, but they were not confirmed by the assay (i.e. false positives) and found evidence in the literature that indeed these genes are likely part of these processes under certain biological conditions. For example, two genes in the top-10 for biofilm formation were annotated with the biofilm pathway in KEGG (more detailed discussion and interpretation can be found in Supplementary Material S17). Interestingly, these genes received low scores by *PCC* and *MR* (see e.g. Supplementary Fig. S12, Supplementary Material S17).

4 Discussion

4.1 *MLC*

We introduced *MLC*, a metric learning method for building automatic function predictors from a large collection of expression data. *MLC* calculates gene co-expression by assigning GO-term-specific weights to each sample. The weights aim at maximizing the co-expression similarity between genes that are annotated with that GO term. In general, training GO-term specific classifiers (also known as the 'Binary Relevance' approach in the machine learning literature) has the disadvantage that individual classifiers fail to see the 'bigger picture' and cannot exploit the correlations between terms imposed by the ontological structure. Several works on multi-label classification have shown that Binary Relevance performs worse than models that incorporate label correlations (Li and Zhang, 2014; Suzuki *et al.*, 2001; Tanaka *et al.*, 2015). Despite this, we showed that the weight profiles learned by *MLC* do correlate with real biological knowledge, such as semantic similarity in the ontology graph and gene annotation similarity, meaning that our method is powerful enough to capture at least some of the label similarities even though it was not exposed to them. Due to the use of the L1 regularization, *MLC* can also select informative samples by setting the weights of non-informative samples to zero. Moreover, we showed that the samples that are selected come from biological conditions relevant to the GO term in question.

Our method is designed to work well with a GBA approach like the *k*-NN classifier. This classifier assigns a GO term to a test gene if a large enough fraction of its top co-expressed training genes are annotated with that term. To achieve this, *MLC* tries to maximize the difference between the average co-expression between gene pairs that are both annotated with the GO term of interest ('*p*–*p*' pairs) and the average co-expression between gene pairs only one of which is annotated with the term ('*p*–*n*' pairs). During the training phase, our model ignores gene pairs where neither gene has the term of interest ('*n*–*n*' pairs). Such pairs could either include two genes that have common GO annotations, but different from the GO term of interest or two genes with completely different annotations. For the first case, one might be tempted to think that the co-expression of such pairs should be high. However, if their common function is different from the term of interest, it is likely that they are correlated for another set of samples than the one related to the GO term of interest and, consequently, are thus uninformative for that GO term. For the second type of '*n*–*n*' pairs, the ones that share no annotations whatsoever, it might make sense to want their co-expression to be 0, as they are expected to be dissimilar over any set of samples. However, we decided to ignore these pairs as they do not add any term-specific information, so it is not clear how they will affect the identification of samples specifically relevant for a specific term. This might be problematic as for a negative test gene (i.e. a gene that should not be annotated with the GO term of interest) we cannot exclude that it can be as highly co-expressed to positive as to negative genes, because we did not tune the co-expression values for '*n*–*n*' pairs. For very frequent terms with a lot of positive training genes this leads to a lot of false positive predictions, which might explain the poor performance of *MLC* for frequent terms.

The similarity function that we used as a basis for *MLC* is the weighted inner product (S_w). We chose this measure because its unweighted version is identical to the unweighted *PCC* for centered and scaled data, but it has a simpler form which eases the computational burden. The weighted versions of the inner product and *PCC* are no longer identical, as the data are no longer scaled after weighing the samples. This has as side-effect that the similarity functions that *MLC* learns are not necessarily in the range [–1, 1], like the *PCC*. In most cases, their range is much narrower as can be seen in Figure 2b for GO: 1903047. Also, because of the range differences, it is not trivial to compare the similarity of two genes across different GO terms. For the purpose of classification with the *k*-NN classifier, however, the range of the metric is insignificant (only the relevant rankings are important to find the proper neighborhood).

Our model is more general and not restricted to only the inner product, though. The main idea is to maximize the difference between the similarity of *p*–*p* and *p*–*n* pairs. This is done by maximizing the *t*-statistic between the two distributions of similarities. This means that *MLC* can also be applied to any measure of similarity such as the weighted *PCC*, weighted Spearman correlation, Euclidean distance etc. Regardless of the chosen metric, the two classes ('*p*–*p*' and '*p*–*n*') do not meet the assumptions for applying Student's *t*-test, as the similarity values are neither normally distributed nor independent. This is not an issue, though, because we do not use the *t*-statistic to compute a *P*-value (exploit that the *t*-statistic is distributed according the Student's *t*-distribution under these assumptions), but only to quantify the class separability (Theodoridis and Koutroumbas, 2008). Equivalently, we could have used any other measure of class separability, for instance the Fisher Discriminant Ratio (Fisher, 1936) or the Davies-Bouldin index (Davies and Bouldin, 1979).

4.2 Comparison to related methods

Our work validates the observation that *PCC* is not the optimal co-expression measure for AFP. The *MR* attempts to obtain more robust and noise-free co-expression values by converting the *PCC*

values into ranks and averaging the reciprocal rankings of two genes (Obayashi *et al.*, 2018). *MLC* takes a fundamentally different approach, operating on the sample level rather than the correlation level. First and foremost, as we mentioned above, it removes samples that do not help at discriminating between genes that do or do not perform a certain function. With that *MLC* gives insight into which samples are important for a given GO term, which subsequently can be used to investigate the expression patterns of the GO term related genes across these samples. Weighing samples differently can also be viewed as a way of denoising. For example, it can compensate for the issue that an expression change of one unit has a different meaning in different samples due to technical variations, such as differences in sequencing depth or sample preparations. Our results have shown that *MLC* is more beneficial than the *MR* approach for the more specific—and arguably more useful—GO terms.

A similar method to *MLC* is *GAAWGEFA*, which learns a weight for each sample in a dataset and then applies a weighted Pearson correlation. There are two fundamental differences between the two methods. Firstly, *GAAWGEFA* aims at good protein-centric performance, i.e. it tries to do well on average for all genes and therefore learns only one set of sample weights. On the other hand, *MLC* aims at maximizing the performance for each GO term individually. Secondly, *GAAWGEFA* learns the weights using a genetic algorithm. For *MLC*, we used the inner product, which allowed us to have a simple optimization problem that can be solved very efficiently. Even though *MLC* has to be run for each term separately, it is still 67% faster than *GAAWGEFA* and, unlike *GAAWGEFA*, runs for different GO terms can be carried out in parallel to achieve even greater speed-up. Next to those differences, *MLC* makes more accurate predictions for rarer terms and provides interpretability of the predictions by examining the term-specific sample weight distributions.

Furthermore, in the context of selecting expression samples a related technique is biclustering. Biclustering is an umbrella term for a diverse set of algorithms that simultaneously select subsets of genes and samples, so that the genes in the same subset (bicluster) have similar expression to each other within the samples of that bicluster. It is typically expected that each bicluster reflects a biological process and that makes the rationale of *MLC* appear similar to a biclustering approach. Although both approaches make use of sample selection and aim at discovering genes involved in the same biological processes, they are fundamentally different in the sense that *MLC* is supervised and biclustering unsupervised. Biclustering does not make use of GO annotations, but only of the expression matrix. Often, observing enrichment of certain GO terms or KEGG pathways in the genes of biclusters is one of the ways to validate a biclustering result (Santamaría *et al.*, 2007). On the other hand, *MLC* starts with a set of genes whose GO annotations are known (or at least partly known) and tries to use the expression matrix in order to identify which of the remaining genes participate in a particular biological process by defining a co-expression measure specific to that process.

4.3 Possible extensions

MLC learns the sample weights automatically from the available data and does not rely on information about the samples' biological condition or tissue. As curation efforts increase and the amount of well-annotated data in public databases grows larger with time, in the future it might be useful to extend *MLC* to incorporate such knowledge. A possible way to do that would be a group LASSO approach (Yuan and Lin, 2006). Group LASSO uses predefined groups of samples and forces the weights of all samples in a group to be equal. Each such group could contain technical and biological replicates, samples from the same tissue or samples from similar knockout experiments and perturbations.

A disadvantage of *MLC* is the fact that it does not account for the possibility that genes that show exactly opposite expression patterns (i.e. genes with large negative correlation) might also be involved in the same biological process. In fact, negative correlations are penalized, as our model explicitly tries to force the signed similarities of *p–p* pairs to be larger than those of the *p–n* pairs. In Figure 2b, we see that large negative *PCC* values are scarce within our dataset, implying that we might not suffer a lot from ignoring negative similarities, at least when considering the *PCC* as a method. The effect might be larger for our *MLC* approach though, which selects a subset of the samples, as in this smaller set negative correlations might be more frequent.

To handle this short-coming, one could directly use the absolute value of the weighted co-expression in the model. Doing this, one is faced with an additional challenge, namely that the absolute value is not differentiable at 0. This can be overcome by approximating the absolute value with a smooth function, such as $\sqrt{x^2+\epsilon}$, where ϵ is a small positive number (Ramírez *et al.*, 2013), but this makes the co-expression function non-linear and the calculation of its derivative with respect to $\mathbf{w}$ more costly. More importantly, it makes the optimization problem more difficult, as it adds an extra non-linearity to an already non-convex objective function, meaning that it might be harder to find a good solution for the weights in this problem.

One could also think of alternative formulations of the objective function that would accommodate absolute co-expression values more easily. For instance, it would be possible to minimize the squared difference between the weighted absolute correlations and a target value (e.g. 0 for *p–n* pairs and 1 for *p–p* pairs). Another possibility would be to use a triplet loss, which has been successfully used in image retrieval (Husain *et al.*, 2019). In the triplet loss, we look at sets of three genes at a time instead of two: two positive genes (p_1, p_2) and one negative (n_1). Then, we maximize the difference $S_w(p_1, p_2) - S_w(p_1, n_1)$, where S_w is in this case the absolute weighted similarity.

Finally, in this work, we applied *MLC* on finding candidate genes for GO terms from the BPO. However, it can be useful for any gene annotation problem that can be solved with expression data, such as finding members of KEGG pathways or genes that are likely to influence a given phenotypic trait. As *MLC* is computationally efficient, it can easily be applied to a large number of different terms/phenotypes, offering state-of-the-art performance with the added benefit of allowing users to understand which parts of the dataset influence the predictions.

Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable suggestions.

Funding

This work was supported by Keygene N.V., a crop innovation company in the Netherlands.

Conflict of Interest: none declared.

References

Ashburner,M. *et al.* (2000) Gene Ontology: tool for the unification of biology. *Nat. Genet.*, **25**, 25–29.

Bellet,A. *et al.* (2013) A survey on metric learning for feature vectors and structured data. *arxiv*, **1306.6709**.

Byrd,R.H. *et al.* (1995) A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, **16**, 1190–1208.

Chen,D. *et al.* (2018) Architecture of gene regulatory networks controlling flower development in *Arabidopsis thaliana*. *Nat. Commun.*, **9**, 4534.

Clark,W.T. and Radivojac,P. (2013) Information-theoretic evaluation of predicted ontological annotations. *Bioinformatics*, **29**, i53–i61.

Clough,E. and Barrett,T. (2016) The gene expression omnibus database. *Methods Mol. Biol.*, **1418**, 93–110.

Cozzetto,D. *et al.* (2013) Protein function prediction by massive integration of evolutionary analyses and multiple data sources. *BMC Bioinformatics*, **14**, S1.

Davies,D.L. and Bouldin,D.W. (1979) A cluster separation measure. *IEEE Trans. Pattern Anal. Machine Intel.*, **PAMI-1**, 224–227.

Fisher,R.A. (1936) The use of multiple measurements in taxonomic problems. *Ann. Eugenics*, **7**, 179–188.

Hu,L.-Y. *et al.* (2016) The distance function effect on k-nearest neighbor classification for medical datasets. *Springerplus*, **5**, 1304.

Husain,S.S. *et al.* (2019) ACTNET: end-to-end learning of feature activations and multi-stream aggregation for effective instance image retrieval. *CoRR*, **abs/1907.05794**.

Jaskowiak,P. *et al.* (2012) Evaluating correlation coefficients for clustering gene expression profiles of cancer. In: de Souto,M.C. and Kann,M.G. (eds.) *Advances in Bioinformatics and Computational Biology. BSB 2012. Lecture Notes in Computer Science*, Vol. 7409. Springer, Berlin, Heidelberg, pp. 120–131.

Jiang,Y. *et al.* (2016a) An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol.*, **17**, 184.

Jiang,Z. *et al.* (2016b) Differential coexpression analysis reveals extensive rewiring of arabidopsis gene coexpression in response to pseudomonas syringae infection. *Sci. Rep.*, **6**. doi: 10.1038/srep35064.

Johnson,W.E. *et al.* (2007) Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics*, **8**, 118–127.

Kavanová,M. *et al.* (2006) Phosphorus deficiency decreases cell division and elongation in grass leaves. *Plant Physiol.*, **141**, 766–775.

Lan,L. *et al.* (2013) MS-kNN: protein function prediction by integrating multiple data sources. *BMC Bioinformatics*, **14 (Suppl. 3)**, S8.

Li,Y.-K. and Zhang,M.-L. (2014) Enhancing binary relevance for multi-label learning with controlled label correlations exploitation. In: Pham,D.-N. and Park,S.-B. (eds.) *PRICAI 2014: Trends in Artificial Intelligence*. Springer International Publishing, Cham, pp. 91–103.

Muller,R. *et al.* (2008) The receptor kinase CORYNE of Arabidopsis transmits the stem cell-limiting signal CLAVATA3 independently of CLAVATA1. *Plant Cell*, **20**, 934–946.

Obayashi,T. *et al.* (2018) ATTED-II in 2018: a plant coexpression database based on investigation of the statistical property of the mutual rank index. *Plant Cell Physiol.*, **59**, e3.

Parkinson,H. *et al.* (2007) ArrayExpress–A public database of microarray experiments and gene expression profiles. *Nucleic Acids Res.*, **35**, D747.

Petryszak,R. *et al.* (2017) The RNASeq-er API–a gateway to systematically updated analysis of public RNA-Seq data. *Bioinformatics (March)*, **33**, 1–3.

Ramírez,C. *et al.* (2013) $\sqrt{x^2+\mu}$ is the most computationally efficient smooth approximation to $|x|$: a proof. *J. Uncertain Syst.*, **8**, 205–10.

Ray,S.S. and Misra,S. (2019) Genetic algorithm for assigning weights to gene expressions using functional annotations. *Comput. Biol. Med.*, **104**, 149.

Resnik,P. (1995) Using information content to evaluate semantic similarity in a taxonomy. In: Mellish,C.S. (ed.), *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1 (IJCAI'95)*, Morgan Kaufmann Publishers Inc., San Francisco, CA, pp. 448–453.

Santamaría,R. *et al.* (2007) Methods to bicluster validation and comparison in microarray data. In: Yin,H. *et al.*, (eds.) *Intelligent Data Engineering and Automated Learning–IDEAL 2007*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 780–789.

Suzuki,E. *et al.* (2001) Bloomy decision tree for multi-objective classification. In: De Raedt,L. and Siebes,A. (eds.) *Principles of Data Mining and Knowledge Discovery*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 436–447.

Tan,J. *et al.* (2017) Unsupervised extraction of stable expression signatures from public compendia with an ensemble of neural networks. *Cell Syst.*, **5**, 63–71.

Tanaka,E.A. *et al.* (2015) A multi-label approach using binary relevance and decision trees applied to functional genomics. *J. Biomed. Inform.*, **54**, 85–95.

Theodoridis,S. and Koutroumbas,K. (2008) *Pattern Recognition*. Academic Press, Orlando, FL.

Tibshirani,R. (1996) Regression selection and shrinkage via the Lasso. *J. Royal Stat. Soc. B* **58**, 267–88.

Varma,S. and Simon,R. (2006) Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, **7**, 91.

Vaucheret,H. *et al.* (2004) The action of argonaute1 in the mirna pathway and its regulation by the mirna pathway are crucial for plant development. *Genes Dev.*, **18**, 1187–1197.

Vázquez-Ramos,J.M. and de la Paz Sánchez,M. (2003) The cell cycle and seed germination. *Seed Sci. Res.*, **13**, 113–130.

Xu,B. *et al.* (2008) Arabidopsis genes AS1, AS2, and JAG negatively regulate boundary-specifying genes to promote sepal and petal development. *Plant Physiol.*, **146**, 566–575.

Xu,Y. *et al.* (2017) Multi-instance metric transfer learning for genome-wide protein function prediction. *Sci. Rep.*, **7**. doi: 10.1038/srep41831.

Yuan,M. and Lin,Y. (2006) Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **68**, 49–67.

Zhang,L. *et al.* (2017) Hierarchical multi-label classification using fully associative ensemble learning. *Pattern Recogn.*, **70**, 89–103.

Zhou,N. *et al.* (2019) The cafa challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *bioRxiv*. doi: 10.1101/653105.