TopoFun: a machine learning method to improve the functional similarity of gene co-expression modules

Ali Janbain, Christelle Reynes, Zainab Assaghir, Hassan Zeineddine, Robert Sabatier, Laurent Journot

Published online 8 November 2021

NAR Genomics and Bioinformatics, 2021, Vol. 3, No. 4 **1**
https://doi.org/10.1093/nargab/lqab103

TopoFun: a machine learning method to improve the functional similarity of gene co-expression modules

Ali Janbain[1,2], Christelle Reynès[1], Zainab Assaghir[2], Hassan Zeineddine[2], Robert Sabatier[1,*] and Laurent Journot[1,3,*]

[1]IGF, Univ Montpellier, CNRS, INSERM, Montpellier 34094, France, [2]Applied Mathematics Department, Lebanese University, Beirut 1003, Lebanon and [3]MGX, Univ Montpellier, CNRS, INSERM, Montpellier 34094, France

Received March 16, 2021; Revised September 22, 2021; Editorial Decision October 11, 2021; Accepted October 13, 2021

ABSTRACT

A comprehensive, accurate functional annotation of genes is key to systems-level approaches. As functionally related genes tend to be co-expressed, one possible approach to identify functional modules or supplement existing gene annotations is to analyse gene co-expression. We describe TopoFun, a machine learning method that combines topological and functional information to improve the functional similarity of gene co-expression modules. Using LASSO, we selected topological descriptors that discriminated modules made of functionally related genes and random modules. Using the selected topological descriptors, we performed linear discriminant analysis to construct a topological score that predicted the type of a module, random-like or functional-like. We combined the topological score with a functional similarity score in a fitness function that we used in a genetic algorithm to explore the co-expression network. To illustrate the use of TopoFun, we started from a subset of the Gene Ontology Biological Processes (GO-BPs) and showed that TopoFun efficiently retrieved genes that we omitted, and aggregated a number of novel genes to the initial GO-BP while improving module topology and functional similarity. Using an independent protein-protein interaction database, we confirmed that the novel genes gathered by TopoFun were functionally related to the original gene set.

INTRODUCTION

In model organisms, geneticists produced a large amount of functional data that attributed one or more functions to many genes. Yet, numerous human, murine, fly, worm, arabidopsis, and yeast genes still have no or sparse functional annotations (1); the situation is obviously worse in non-model organisms. The available information on gene function was categorized in projects such as the Gene Ontology (GO) (2), the Kyoto Encyclopaedia of Genes and Genomes (KEGG) (3), and the Reactome pathway database (4). These projects are key to genome-wide approaches as they help organize the thousands of genes harboured by prokaryote and eukaryote genomes into functional 'pathways' or 'biological processes'. The resulting gene classification is widely used to infer functional knowledge from omics data, *e.g.* by testing the over-representation of GO terms/KEGG pathways/Reactome pathways among differentially expressed genes in transcriptomic data. It is then crucial that the gene functional annotations are as accurate and comprehensive as possible.

When no or few experimental data were available, computer biologists developed various methods to infer gene function from genome scale data. One possible approach is to analyse gene co-expression as co-expressed genes tend to be functionally related (5–8). The first step in this process is to define a metric that captures the similarity between gene expression profiles for any gene pair across a series of biological samples. Euclidian distance, Pearson's correlation, Mutual Rank, and Bayesian metric have all been used with some pros and cons (9). The second step is to set a threshold value for the selected metric to classify gene pairs as 'co-expressed' and 'non-co-expressed'. The resulting data are represented as a high-dimensional graph in which nodes denote genes and (weighted) edges denote co-expression. As functionally related genes tend to be co-expressed, genes involved in the same function tend to cluster in the co-expression network and form a so-called functional module. The identification of such modules is critical to systems approaches (10).

Community detection is a longstanding question in network science (11,12). Most methods, including the most popular in biology (13), use topological information to find groups of nodes that are densely connected relative to the rest of the network (14). These methods are difficult to benchmark as ground truth is impossible to know in real-world networks and the use of node-associated metadata as

*To whom correspondence should be addressed. Tel: +33 434359241; Email: laurent.journot@igf.cnrs.fr
Correspondence may also be addressed to Robert Sabatier. Tel: +33 411759680; Email: robert.sabatier@umontpellier.fr

a proxy is questionable (15,16). Other methods are aimed at a slightly different task, *i.e.* the identification of 'active modules'. They combine network topology and molecular profiles, *e.g.* transcriptomic data or disease state, to identify connected sub-networks that show significant changes under different conditions (10,17). The recent Disease Module Identification DREAM challenge (https://synapse.org/modulechallenge) highlighted the complexity of the task, the limits of the available methods, and the lack of improvement using multi-layered networks (18). Given these limitations, some methods were developed to incorporate biological knowledge, *i.e.* gene functional annotations, in different ways. The first attempts were to use functional annotations with gene expression data after the clustering step as in EXPANDER (19) or during the clustering step as in PANA (20) and the method proposed by Leale *et al.* (21). Functional annotations were also used in gene co-expression network analysis as in EGAD (22). Recently, a machine learning approach, 'Multi-Features Relatedness' (23), was developed; it combined co-expression information, functional annotations, protein-protein interactions, and textual information. It was limited to study the multi-feature relationship between pairs of genes, not multi-gene functional modules. Finally, the most classical approach was to use functional annotations *a posteriori* to assign one or more function(s) to the different topologically defined clusters using classical network algorithms such as WGCNA (13), CoExpNets (24), ARACNE (25), GeneMania (26), etc.

In the present work, we used a different strategy and favoured a gene set-centred approach. Starting from a seed gene set, we aimed at exploring co-expression networks to test whether these genes, or a subset thereof, were significantly co-expressed, and to identify novel genes significantly co-expressed with the seed genes. When combined with functional annotations, such a method would be of value to contextualize sets of genes called differentially expressed in transcriptomic data or identified in a GWAS. To perform a systematic network exploration, the primary set of genes could be made of one specific gene and its *n*th-degree neighbours. Alternatively, starting from a set of genes known to be functionally related, such a method would help clarify the pathways (made of co-expressed genes) in which they operate and find novel functionally related genes. In a previous work, we attempted to discriminate random modules (RMs) from functional modules (FMs). Random modules were made of genes randomly sampled among the genes in the co-expression network. A functional module was defined according to Hartwell and colleagues (27) as 'a group of genes or their products which are related by one or more genetic or cellular interactions, *e.g.* co-regulation, co-expression or membership of a protein complex, of a metabolic or signalling pathway or of a cellular aggregate (e.g. chaperone, ribosome, protein transport facilitator, etc.)'. We used GO Biological Processes as prototypic FMs and computed two topological descriptors (28). The average mutual rank was not predictive of a module class, random or functional; in contrast, the average degree largely discriminated the two classes of modules (Supplementary Figure S1). In the present work, we generalized this approach and performed a systematic study of modules topological descriptors. We used LASSO (least absolute shrinkage and selection operator) (29) and LDA (linear discriminant analysis) (30) to find a linear combination of the relevant descriptors that optimally separated FMs and RMs. We combined this topological score with a functional score based on the work of Wang and colleagues (31) in a fitness function implemented in a genetic algorithm (GA), which we named TopoFun. To illustrate TopoFun capabilities, we applied the method to a subset of GO-BPs and showed that it was able to discover novel functionally related genes.

MATERIALS AND METHODS

Co-expression data and metrics

Mouse co-expression data were obtained from COXPRESdb Mmu.c3-1 (32). COXPRESdb uses Mutual Rank (MR) to measure co-expression. The Mutual Rank (MR) of genes g_i and g_j is defined as follows: the list of genes co-expressed with g_i, respectively g_j, is sorted according to the value of the Pearson's correlation coefficient (PCC). The rank of g_j, respectively g_i, is recorded. The MR is the geometric mean of the two ranks; the lower the MR, the higher the co-expression of the 2 genes. According to Obayashi and Kinoshita, MR showed better performance than PCC in GO prediction (33). If the MR of 2 genes is lower than a given threshold μ, they are considered as co-expressed. Hence, the COXPRESdb network can be fully described by the following binary adjacency matrix A:

$$a_{ij} = \begin{cases} 0 \text{ if } \mathrm{MR}(ij) > \mu \\ 1 \text{ if } \mathrm{MR}(ij) \leq \mu \end{cases} \quad (1)$$

We previously suggested that a reasonable value for μ is 1200 (28). Here, we performed a more systematic study and tested 12 values for μ ranging from 300 to 2000 (Supplementary File, section 2). We generated FMs and RMs for each μ value and calculated all the descriptors of the resulting modules. We used LDA, for each μ, to find a linear combination of the descriptors that optimally separated FMs and RMs. We performed cross-validation to estimate the performance and the error of the model, and noted that the error stabilized for $\mu = 1200$ (Supplementary Figure S2). We used $\mu = 1200$ for the rest of the work.

Topological descriptors and topological score

Topological descriptors, also named 'topological parameters', of a network are metrics that characterize a facet of a network topology such as the density of edges, the length of the shortest path (hop count) between two nodes, the density of edges between the neighbours of a given gene… (34–36).

Selecting the relevant topological descriptors. To construct the database to train the machine learning model, we generated all, *i.e.* 978, GO-BPs (FMs) that comprised 20–500 genes; we also generated 1000 RMs that contained 20–500 randomly sampled genes and whose size distribution is comparable to that of the GO-BPs (Supplementary Figure S3, Size). We computed 12 descriptors (definitions are given in Supplementary File, section 3) from the 1978 modules and summarized the results in Supplementary Table S1.

We had a set of modules D = (x_1, y_1), …, $(x_{1,978}, y_{1,978})$ such that for every module i, the vector $x_i = (x_i1, \ldots, x_i12)$ comprised the 12 descriptors of module i. The vector of binary responses y_i informed about the module class, 'FM' or 'RM'. The logistic regression model assumed that

$$P(y_i = \mathrm{FM}|x_i) = \frac{1}{1 + e^{-x_i'\alpha}}, \quad (2)$$

where $\alpha \in \mathbb{R}^{12}$ was the vector of the regression coefficients. We used this matrix to learn a LASSO model and select the predictor variables (relevant descriptors). The LASSO estimated the regression coefficients by minimizing the least squares quadratic criterion under a constraint on the sum of the absolute values of the coefficients or, equivalently, by minimizing the quadratic criterion penalized by the norm L1 of the coefficients. This technique was aimed at removing the redundant or irrelevant descriptors without much loss of information. The LASSO method selected 6 relevant descriptors: Shortest Path, Betweenness Centrality, Average Degree, Inverse Centrality, Clustering Coefficient, and Size. The distribution of each relevant descriptor for RMs and FMs is shown in Supplementary Figure S3. As expected, we observed that the average Betweenness Centrality, the Average Degree and the average Clustering Coefficient were higher in FMs than in RMs, confirming that FMs tended to be denser and more compact than RMs.

Defining the topological score of a gene set. We then applied LDA to find a linear combination of the relevant descriptors that optimally discriminated FMs and RMs (37). We extracted a training set from Supplementary Table S1 by keeping the columns for the 6 relevant descriptors and the response vector. As two classes were to be predicted, only one discriminant axis could be built; we denoted $Score_{Topo}$ the coordinate of each module on this axis. The coefficients of the 6 descriptors were Shortest Path, –0.67294573; Betweenness Centrality, 32.37896628; Average Degree, 0.04082761; Inverse Centrality, 9.15250821; Clustering Coefficient, 0.19094753; Size, –0.00845438. To assess the contribution of each descriptor to the LDA model, we calculated the correlation between $Score_{Topo}$ and each of the six descriptors; Shortest Path, 0.12; Betweenness Centrality, 0.54; Average Degree, 0.23; Inverse Centrality, 0.76; Clustering Coefficient, 0.44; Size, 0.24.

Data in Table 1 showed that 973 out of 1000 RMs and 847 out of 978 FMs were correctly classified, corresponding to an overall classification error rate of 7.99%. Interestingly, 2.8% of the RMs and 13.4% of the FMs were misclassified. To further illustrate the classification performance, we show the distribution of $Score_{Topo}$ for RMs and FMs in Figure 1. We used two-fold cross-validation to estimate the prediction performance of the model when applied to new data (37). Briefly, we randomly split the training set into two equal subsets s_0 and s_1. We trained a LDA model on s_0 and validated the model on s_1, followed by training on s_1 and validation on s_0. The process was repeated 1000 times and the average classification error rate was 8.36%. The corresponding receiver operating characteristic (ROC) curves and the distribution of the area under the curves (AUCs) are shown on Figure 2. In Figure 2B, the histogram obtained through cross-validation showed that all values were above 0.91 and were distributed in a very narrow range demonstrating the stability and robustness of the method through cross-validation.

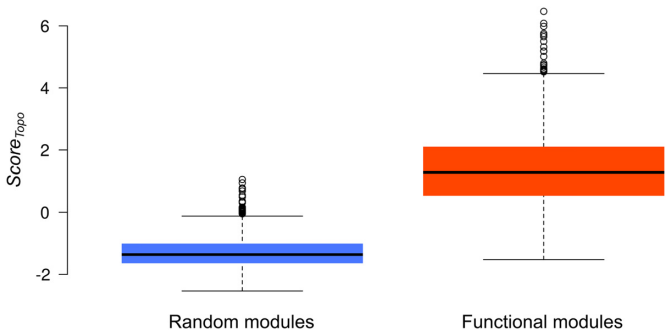

Figure 1. The distribution of $Score_{Topo}$ for random and functional modules. The topological score is a linear combination of the 6 descriptors of module topology selected by the LASSO method, *i.e.* shortest path, betweenness centrality, average degree, inverse centrality, clustering coefficient, and size. It was computed for the 978 Gene Ontology Biological Processes consisting of 20–500 genes and for 1,000 random modules of the same sizes. The $Score_{Topo}$ distributions for random (blue) and functional (red) modules indicated that $Score_{Topo}$ largely differentiated both types of modules.

Table 1. The confusion matrix for the LDA model applied to the training set

		Actual class		
		RM	FM	Total
Predicted class	**RM**	973	131	1,104
	FM	27	847	874
	Total	1,000	978	1,978

We performed linear discriminant analysis of the $Score_{Topo}$ computed for 978 Gene Ontology Biological Processes (FMs, functional modules) consisting of 20–500 genes and 1,000 random modules (RMs) of the same sizes. The table displays the calls of the linear discriminant analysis, indicating that 97.3% of the random modules were correctly classified, and 13.4% of the functional modules were misclassified.

Gene functional similarity score

The correspondence between genes and GO terms was obtained from the gene2go file (version 30) from the NCBI repository (38). We used the method proposed by Wang and colleagues (31), which encodes the semantics of a GO term into a numerical value by summing the semantic contributions of their ancestor terms (including this specific term) in the GO graph.

Defining the gene functional similarity score. Several methods have been developed to measure the functional similarity of genes using GO annotations. Some measure gene functional similarities based solely on the probability of the appearance of GO terms and ignore the semantic relations among these terms in the GO graph. Other methods were proposed to measure the semantic similarity of terms in a specific taxonomy (39–41); they were originally developed for the natural language taxonomies and it is unclear whether they are suitable for measuring the semantic similarity of GO terms. Ideally, the semantics of a GO term (biological meaning) should include the biological meanings

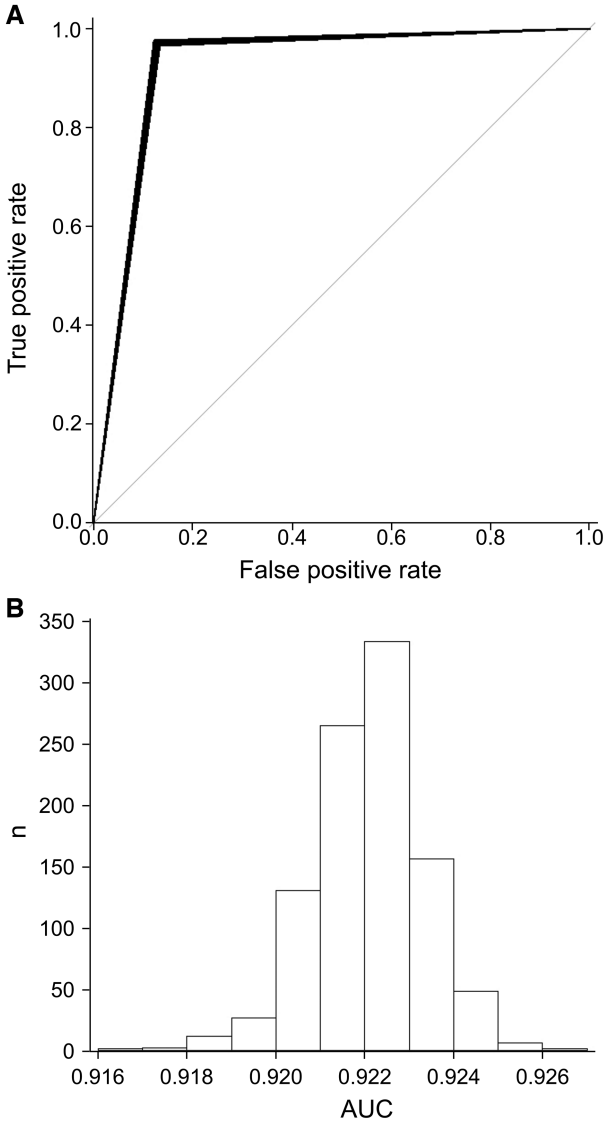

Figure 2. Classification performance of $Score_{Topo}$. (**A**) Receiver Operating Characteristic (ROC) curves of the two-fold cross-validation of $Score_{Topo}$. The training set was split into two equal subsets. A LDA model was trained on one subset and tested on the other, and *vice-versa*. The process was repeated 1,000 times and the 1,000 ROC curves were plotted. The average classification error rate was 8.36%. (**B**) The distribution of the corresponding area under the curves (AUCs).

of all its ancestor terms. The measurement of the semantic similarity of two GO terms should then take into account not only the common ancestors number, but also the locations of these ancestor terms related to the two specific terms in the GO graph. Wang and colleagues (31), encoded the semantic value of a GO term G into a numerical value by aggregating to the semantics of G the contributions of all terms in the directed acyclic graph that includes G. Terms closer to G contribute more to its semantics, while terms farther from G contribute less as they are more general terms. The semantic value of each term is then used to compute the semantic similarity of two GO terms. The semantic similarity between one term *go* and a GO term set $GO = \{go_1, go_2, \ldots, go_k\}$, Sim(*go*, *GO*), is then defined as the maximum semantic similarity between the term *go* and any of the terms in set *GO*. That is

$$\mathrm{Sim}(go,\ GO) = \max_{1\le i\le k}\left(\mathrm{S_{GO}}\left(go,\ go_i\right)\right), \tag{3}$$

where $S_{GO}(go,\ go_i)$ is the semantic similarity of the two terms *go* and go_i. Given two genes g_i and g_j annotated by GO term sets $GO_i = \{go_{i1}, go_{i2}, \ldots, go_{im}\}$ and $GO_j = \{go_{j1}, go_{j2}, \ldots, go_{jn}\}$ respectively, Wang and colleagues (31) defined their functional similarity as

$$\mathrm{Sim}\ (g_i,\ g_j) = \frac{\sum_{1\le k\le m}\mathrm{Sim}\left(go_{ik},\ GO_j\right) + \sum_{1\le k\le n}\mathrm{Sim}\left(go_{jk},\ GO_i\right)}{m+n} \tag{4}$$

We used the *geneSim* function, with *measure* = 'Wang', of the *GOSemSim* package to compute the semantic similarity of two genes. We computed β, a matrix of functional similarity between all the genes in COXPRESdb, based on this method. For any module, denoted M, composed of n genes, we extracted the sub-matrix of β representing the similarity between all pairs of genes in the module. This matrix was symmetric. We defined

$$Score_{Fun} = \frac{\sum_{i>j}\mathrm{Sim}\left(g_i, g_j\right)}{n\left(n-1\right)} \tag{5}$$

$Score_{Fun}$ was a number between 0 and 1; the closer to 1, the more the module was composed of functionally similar genes.

Implementing the TopoFun genetic algorithm

Starting from a gene set, we aimed at discovering novel genes co-expressed with members of that gene set. The number of candidate genes was far too large for a systematic scanning. We therefore implemented the scanning process through a genetic algorithm (GA), which involved the construction of a fitness function. In the GA, M_0 is the original gene set, M_1 is one of the 500 gene sets produced by TopoFun at each iteration and M_f is the final, ranked #1 gene set after TopoFun converged.

Fitness function. In order to rank modules, we designed a fitness function that quantified the quality of these modules with respect to both co-expression and shared functions. The Supplementary File, section 5, describes the different steps that led to the definition of the fitness function *FF*. This function, to be maximized, combined topological and functional information for any new module M_1, derived through addition and deletion of genes from a known module M_0:

$$FF_{M_0}(M_1) = \sqrt{\delta_0 + \delta_1 \times Score_{Topo}(M_1)} \times Score_{Fun}(M_1) \times \frac{|M_1 \cap M_0|}{|M_0|} \tag{6}$$

$Score_{Topo}$ favoured modules with FM-like topology. $Score_{Fun}$ favoured modules composed of functionally related genes. $|M_1 \cap M_0|$ was the number of genes belonging to M_0 and M_1 and $|M_0|$ the number of genes in M_0. The latter term was aimed at 'anchoring' the novel module around

M_0; without this term, the module was likely to drift towards a neighbour module, mostly independent from M_0 but with better topological and functional properties.

When maximizing *FF*, the three terms should contribute to the quantification of M_1 quality to the same extent. However, their range were very different; the third term obviously belonged to [0,1], and so did $Score_{Fun}$. In contrast, $Score_{Topo}$ ranged from –2.535 to +6.466 on the learning data set (Figures 1 and3). We introduced δ_0 (= 0.281608) and δ_1 (= 0.1110948) to rescale $Score_{Topo}$ using the learning dataset so that the RM with the lowest $Score_{Topo}$ was set to 0 and the FM with highest $Score_{Topo}$ was set to 1.

Genetic algorithm. GA are inspired by natural selection (42), and are useful in complex optimization problems (43,44). Here, we used a GA to identify 'optimized' modules by maximizing *FF*. We initialized the algorithm with a population constituted of different modules. At each iteration, the population evolved according to three operators described in the next paragraphs: crossover, mutation and selection. Selection was crucial to keep the fittest modules. Mutation and crossover were run independently of the selection process.

Initialization. For each module to test, we used a first-generation population of 500 modules, which each contained up to 500 genes. Given M_0 such that $5 \leq |M_0| \leq 500$, the first generation-population was composed of

- 300 modules identical to M_0,
- 100 modules composed of the largest clique of M_0 and 80% of the genes of M_0 in the complement to the largest clique, and
- 100 modules composed of all genes in M_0 and $(500 - |M_0|)$ randomly sampled genes (not in M_0).

Selection. We implemented the selection based on the fitness values as in Reeves and Rowe (45). We ranked the modules according to their fitness value, the best one having the highest rank. We then calculated a probability to keep a module in the next generation. A module's probability was proportional to its rank, so that the sum of probabilities over modules summed to 1, and that the best module was twice as likely to be selected as the module with median rank. We also applied elitism to keep the best module of each generation in the next one.

Crossover. At each generation, we produced new combinations of the previously selected modules through single-point crossover (45). We excluded from the crossover the largest clique, which we kept in every module, and applied crossover to 50% of the modules of a generation.

Mutation. We introduced mutations to efficiently explore the solution space and make sure that any point of this space can be reached within a finite number of generations. The possible mutations included substitution, deletion, or addition of one gene. We introduced one mutation per module and the probability of each type of mutation was 0.5 for substitution, 0.375 for deletion and 0.125 for addition. The genes of the largest clique were excluded from the mutation process.

Convergence and stopping criterion. As shown by Bhandari and colleagues (46), two conditions are necessary and sufficient for a GA to converge as the number of iterations goes to infinity:

- The best individual in the population has a fitness value no less than the fitness values of the optimal individual from the previous populations, which we implemented through selection.
- Each solution has a positive probability of going to an optimal individual within a finite number of iterations, which we implemented through mutation.

We assumed that the GA had converged if the best module was identical for 100 generations. We examined the convergence of the GA for 193 M_0 with 50–100 genes, and observed that it consistently converged in <3000 generations; we thus set the default number of iterations to 3000 and stopped the GA when the best solution was identical for 100 generations.

RESULTS

Properties of the topological and functional scores

The present work, and the guilt-by-association approach in general, is based on the premise that frequent gene co-expression predicts functional similarity (47,48). $Score_{Topo}$ describes the topology of the co-expression module and $Score_{Fun}$ is a measure of the functional similarity among a set of genes. If the premise of our work is correct, we should observe no correlation between the two scores for RMs and some correlation for FMs. We plotted $Score_{Topo}$ versus $Score_{Fun}$ for the FMs and the RMs defined above (Figure 3). RMs displayed low $Score_{Fun}$ values confirming they were composed of genes that were not functionally related, as expected for randomly sampled genes. In contrast, FMs had higher $Score_{Fun}$ so that $Score_{Fun}$ almost perfectly discriminated RMs and FMs. $Score_{Topo}$ did not discriminate FMs and RMs as efficiently as $Score_{Fun}$ for reasons that will be discussed in the Discussion section. We observed no correlation of $Score_{\mathrm{Topo}}$ versus $Score_{Fun}$ for RMs (PCC = 0.003), and a correlation for FMs (PCC = 0.364), which confirmed that $Score_{Topo}$ displayed the anticipated properties.

TopoFun gathered known functional relationships

To test the effectiveness of TopoFun, we performed a simulated experiment using GO-BPs as benchmark FMs. We left a number of genes out of different GO-BP modules, ran TopoFun on these 'trimmed' modules, and assessed whether the omitted genes were reinstated. We focused on the set of the 978 GO-BPs that comprised 20–500 genes. To assess whether the topology and size of the GO-BPs affected the effectiveness of TopoFun, we partitioned the 978 modules into 16 classes according to the quartiles of the module size and normalized $Score_{Topo}$ values. The number of GO-BPs in each of the 16 classes is displayed in Supplementary Table S2. We randomly selected five GO-BPs in each of the 16 classes. From each of the 80 selected GO-BPs we generated three sub-modules by keeping 20, 40 or 60% of the

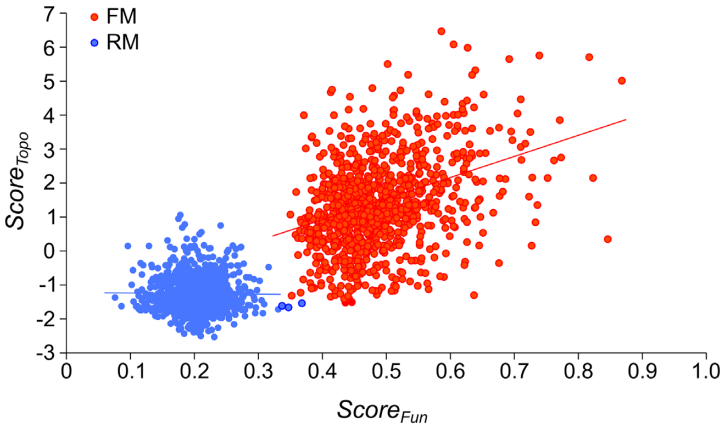

Figure 3. Plot of $Score_{Topo}$ *vs.* $Score_{Fun}$ for random and functional modules. No significant correlation was observed for 1,000 random modules (RM; PCC = 0.0037, *P*-value < 0.9054) in contrast to 978 functional modules (FM; PCC = 0.3639, *P*-value < 2.2×10^{-16}) with gene numbers ranging from 20 to 500.

genes in the largest clique (LC) and all the genes in the complement of the largest clique (LC^C). We generated three additional sub-modules by keeping the LC and 20, 40 or 60% of the genes in the LC^C. We ran TopoFun on the resulting 480 sub-modules and calculated the percentage of omitted genes that were reinstated by TopoFun in the final module.

Genes omitted from the LC were more efficiently reinstated than those from the LC^C (Figure 4A). The percentage of reinstated genes was also more variable for the genes of the LC than for those of the LC^C (Figure 4A), suggesting that genes in the LC were critical in TopoFun. The different behaviours of LC and LC^C genes are also illustrated in Figure 4B. The percentage of LC^C genes reinstated was highly dependent on the percentage of LC^C genes kept; the more LC^C genes were kept in the sub-module, the higher the percentage of LC^C genes were reinstated (Figure 4B, lower panel). In this case, the $Score_{Topo}$ value or the number of genes in M_0 only marginally affected the reinstating rate. In contrast, when genes from the LC were omitted, the size of the M_0 module largely affected the reinstating rate, at least for the 20 and 40% LC sub-modules (Figure 4B, upper panel). This was likely due to the fact that M_0 modules with <25 genes had a LC that often comprised as low as five genes (Supplementary Supplementary Figure S7). Keeping 20% of these five genes resulted in just one gene, which made TopoFun much less efficient. Surprisingly, the $Score_{Topo}$ of the M_0 module was not drastically affecting the reinstating rate but, possibly, for the smallest modules (Figure 4B). In short, these data indicated that TopoFun efficiently discovered novel functionally related genes starting from a functional co-expression module. They also showed that the size, rather than the topology, of the original module was the most critical parameter. One may anticipate that very small modules, *e.g.* with less than 10 genes, may be less efficiently optimized by TopoFun than larger modules.

TopoFun discovered novel functional relationships

To illustrate the performance of TopoFun, we excluded the smallest GO-BPs for reasons mentioned above and focused on the 193 GO-BPs with 50–100 genes (Supplementary Table S3). Because some genes were not in COXPRESdb, the actual gene number ranged from 46 to 99. For each GO-BP, *i.e.* M_0, we ran TopoFun and selected a final, 'optimized' module M_f. Supplementary Table S4 displays the number of genes, the $Score_{Topo}$, and the $Score_{Fun}$ for each M_0 and M_f. We calculated the ratio of each variable for M_f to M_0, and plotted the distribution of these ratios in Figure 5. The final modules comprised substantially more genes (Figure 5, left panel), with improved topology (Figure 5, middle panel) and functional similarity (Figure 5, right panel) than the original modules. This indicated that TopoFun was able to aggregate novel genes to existing GO-BPs while preserving, most often improving, the functional similarity of the constituting genes.

We tested whether the novel genes gathered by TopoFun were functionally relevant using STRING, 'a database of known and predicted protein-protein interactions, including direct (physical) and indirect (functional) associations' (49), unrelated to COXPRESdb. For each GO-BP, we counted the number of edges between the novel genes, *i.e.* $M_0{}^C = M_f - M_0$, and those in the M_0 module. For each M_0, we randomly sampled 10 000 sets of $|M_0{}^C|$ STRING IDs and counted the number of edges between each random set and M_0. Using the empirical cumulative distribution function, we calculated the Benjamini–Hochberg-corrected (50) *P*-value to observe a number of edges as high as that observed for $M_0{}^C$. Of the 193 GO-BPs, 182 had *P*-values < 1×10^{-4}, four had *P*-values < 1×10^{-2}, one had a *P*-value ~0.012, and six were not significant (Figure 6A, Supplementary Table S5). Figure 6B–F illustrates representative STRING networks obtained with 'optimized' modules generated by TopoFun, starting from GO:000082~G1/S transition of mitotic cell cycle (Figure 6B; edge enrichment, 10.6; *P*-value < 1×10^{-4}), GO:0051865~protein autoubiquitination (Figure 6C; edge enrichment, 3.90; *P*-value < 1.10^{-4}), GO:0043401~steroid hormone mediated signaling pathway (Figure 6D; edge enrichment, 1.49; *P*-value < 1×10^{-4}), GO:0010923~negative regulation of phosphatase activity (Figure 6E; edge enrichment, 0.88; *P*-value, ns).

Those modules with a weak edge fold enrichment still proved to be interesting. GO:0043401~steroid hormone mediated signaling pathway (Figure 6D) originally consisted of 53 genes, mostly nuclear receptors for various

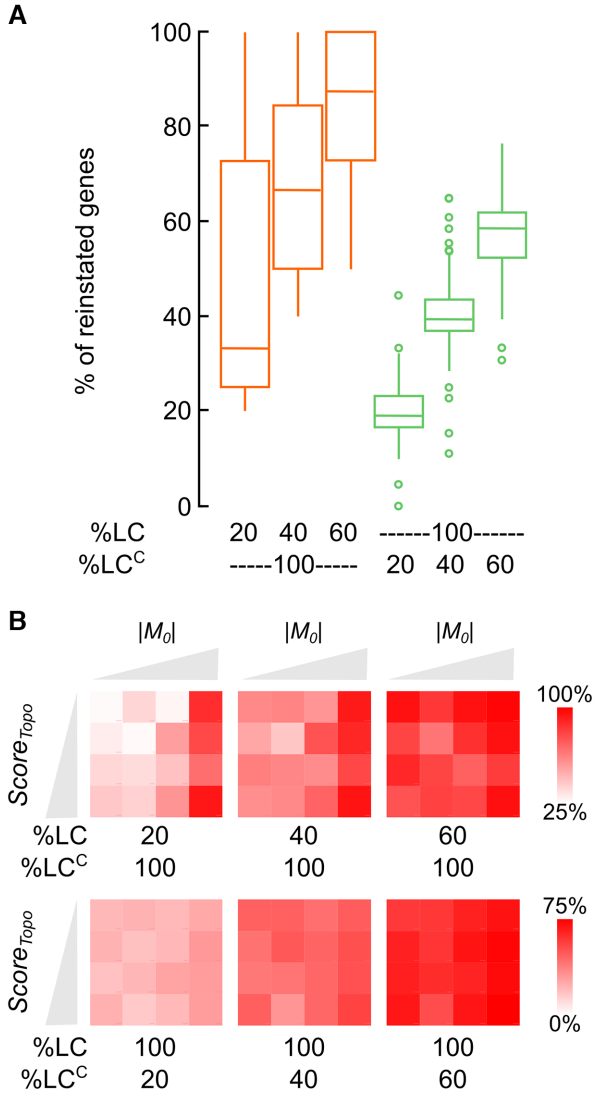

Figure 4. Analysis of TopoFun effectiveness at discovering functionally related genes. We partitioned the 978 GO-BPs with 20–500 genes into 16 classes according to their $Score_{Topo}$ and size ($= |M_0|$) as displayed in Supplementary Table S2. We randomly selected five GO-BPs in each class and generated sub-modules by keeping 20, 40 or 60% of the largest clique (LC) and the full complement to the largest clique (LC^C), or by keeping the LC and 20, 40 or 60% of the LC^C. We ran TopoFun on the 480 sub-modules and calculated the percentage of the genes that were omitted and reinstated by TopoFun. (**A**) Box plots of the distribution of the percentage of reinstated genes for each of the six categories of sub-modules. (**B**) We calculated the means of the percentage of reinstated genes for each of the 16 classes of modules and for each of the six categories of sub-modules, and represented the data as heat maps.

hormones and metabolites such as estrogen, androgens, retinoids, thyroid hormone, vitamin D etc. (Supplementary Table S6). TopoFun preserved the constituting genes and gathered 127 novel genes, which encoded nuclear proteins, including zinc finger proteins, transcription factors, co-activators, co-repressors, chromatin remodelling proteins, RNA binding proteins and helicases (Supplementary Table S6). Given the annotations associated with the genes in GO:0043401 and with the novel genes, we hypothesized that the genes of GO:0043401 represented the nuclear proteins involved in hormone/metabolite binding and the novel genes corresponded to the associated transcription regulatory machinery.

TopoFun identified disjoint co-expressed gene sets in a single GO biological process

Another interesting, atypical case is GO:0007596~blood coagulation (Figure 6F). This GO-BP was the only GO-BP that TopoFun downsized (Figure 5, left panel) and was one of the six modules for which the number of edges between $M_0{}^C$ and M_0 was not significantly increased (edge enrichment, 1.00; P-value = 0.51). This GO-BP originally comprised 71 genes in COXPRESdb; TopoFun excluded 11 genes and added two novel genes. Gene Ontology defines 'blood coagulation' as 'the sequential process in which the multiple coagulation factors of the blood interact, ultimately resulting in the formation of an insoluble fibrin clot'. Most of the genes in this GO-BP were related to the proteolytic cascade involved in clot formation and its regulation: complement components, coagulation factors, serine proteases and inhibitors thereof, fibrinogens, thrombomodulin, thrombin receptors… Most of the core components of the coagulation cascade are produced by the liver, and the placenta during gestation. These genes were retained by TopoFun, confirming that they were co-expressed. Remarkably, the 11 genes that were removed by TopoFun, did not display the same characteristics. These genes (*Ap3b1*/*Hps2*, *Rab27a*, *Lyst*, *Bloc1s6*, *Hps6*/*Bloc2s3*, *Shh*, *Lnpk*, *Dtnbp1*/*Bloc1s8*, *Ano6*, *Hps4*/*Bloc3s2*, *Bloc1s3*) were all annotated 'blood coagulation' by inference from mutant phenotype (IMP). They were then rightfully annotated 'blood coagulation' as they participate in this process at the organismal level. All of them, but *Shh* and *Lnpk*, were annotated with terms related to biogenesis of lysosomal organelles and were ubiquitously/broadly expressed. Of note, mutations of the human orthologues of six of them (*Ap3b1*→*HPS2*, *Bloc1s6*→*HPS9*, *Hps6*→*HPS6*, *Dtnbp1*→*HPS7*, *Hps4*→*HPS4*, *Bloc1s3*→*HPS8*) resulted in the Hermansky-Pudlak syndrome (HPS, hence the gene names), a rare genetic disease characterized by decreased pigmentation (albinism) with visual impairment, and blood platelet dysfunction with prolonged bleeding. The human orthologues of three other genes were also informative. *LYST* is responsible for the Chediak-Higashi syndrome, 'a rare, inherited, complex, immune disorder characterized by reduced pigment in the skin and eyes (oculocutaneous albinism), immune deficiency with an increased susceptibility to infections, and a tendency to bruise and bleed easily'. *ANO6* is responsible for the Scott syndrome, 'a mild platelet-type bleeding disorder characterized by impaired surface exposure of procoagulant phosphatidylserine on platelets and other blood cells' and deficiency of platelet binding of Factor X. *RAB27A* is responsible for type 2 Griscelli syndrome. No bleeding defect is documented for these patients but they are characterized by low levels of platelets and fibrinogen, the fibrin precursor, and light skin. In short, these data suggested that nine of the 11 genes left out by TopoFun were annotated 'blood coagulation' because they function in platelet lysosome formation. We concluded that GO:0007596 comprised (i) genes

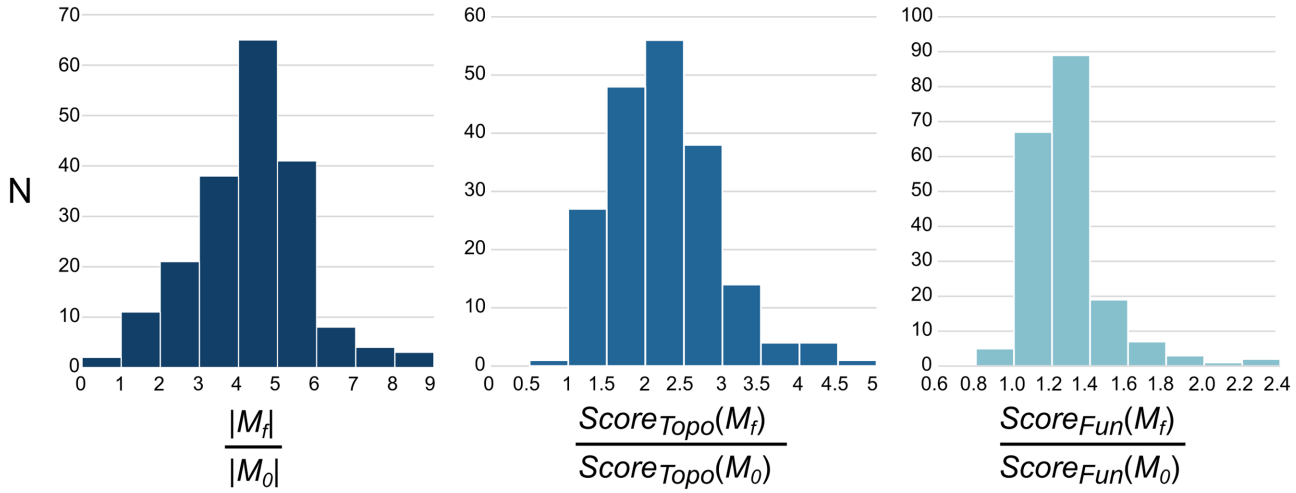

Figure 5. Distribution of the size ratio, $Score_{Topo}$ ratio, and $Score_{Fun}$ ratio. We ran TopoFun on 193 GO-BPs comprising 50–100 genes. For each M_0 (= GO-BP) and M_f (= 'optimized' module), we determined the number of genes, the $Score_{Topo}$ and the $Score_{Fun}$, and plotted the distribution of the ratios of these variables for M_f to M_0.

expressed in the liver that participated in the proteolytic cascade involved in clot formation and (ii) genes expressed in megakaryocytes (the platelet progenitors) that participated in platelet formation and, ultimately, clotting; it is then not surprising that the two categories of genes were not co-expressed. Because the genes of the first category were more numerous in GO:0007596 than those of the second, the largest clique of this BP was composed exclusively of genes of the first category. Because the genetic algorithm used in TopoFun preserved the largest clique of the initial GO-BP at each iteration, genes of the second category were excluded. In the STRING database, the 11 genes left out and the two genes gathered by TopoFun did not display any interaction with the other genes in GO:0007596 or between them (Figure 6F).

TopoFun is not biased towards a limited number of highly connected modules

Because TopoFun favoured highly connected modules, we were concerned that it may frequently converge towards a limited number of highly connected sets of genes that behaved as 'attractors' during the genetic algorithm iterations. We computed the overlap distance (for two sets S_1 and S_2, overlap_distance(S_1, S_2) = 1 – ($|S_1 \cap S_2|$ / min($|S_1|$, $|S_2|$)) for each pair of 'optimized' modules generated by TopoFun. We used this distance to perform hierarchical clustering of the 'optimized' modules and displayed the results as a heat map (Figure 7). The 193 modules were grouped into 38 clusters, of which five included 38, 34, 27, 20 and 17 modules.

The Gene Ontology terms are organized in a hierarchical tree where one given term is a subclass of a more general term. The genes of a given term are then part of the genes constituting a term of higher level in the tree. The fact that some 'optimized' module have a similar gene composition may then result from the interlinked structure of the GO terms. If this was the case, we expected that the more functionally similar the initial modules were, the more the 'optimized' modules should be. We computed the functional similarity of each pair of M_0 modules in each of the 38 clusters, did the same for the M_f modules, and plotted the similarities of the M_0 and M_f pairs for each cluster (Figure 8). We observed for each of the five most populated clusters that the functional similarities of the M_f modules were correlated to that of the M_0 modules; PCC values were 0.75, 0.78, 0.62, 0.78 and 0.64 for clusters # 2, 3, 6, 7, 19, respectively. We concluded that the observed similarity of some M_f modules was not an artefact and was related to their original similarity and to the interlinked relationships of the GO terms.

Comparison to classical network algorithms

A benchmark study comparing TopoFun to classical network algorithms such as WGCNA (13) and CoExpNets (24), ARACNE (25), GeneMania (26) etc. is highly desirable but practically not possible as TopoFun is gene set-centred. TopoFun could theoretically identify modules systematically, *e.g.* starting from each gene and its *n*th-degree neighbours. It is however not practically possible due to the computation time of the GA (one iteration takes about 3.5 s on a laptop). As it is now, TopoFun is better suited to 'optimize' modules identified by classical network algorithms. Accordingly, we ran WGCNA (with *minClustSize* set to its default value, 20) on COXPRESdb Mmu-c3-1 and identified 253 modules comprising 20–701 genes (median, 57; iqr, 59). For a fair comparison, we focused, as we did previously with GO-BPs, on the 84 WGCNA modules with 50–100 genes. We ran TopoFun on these 84 M_0 modules and obtained the corresponding M_f modules (Supplementary Table S7). We looked at the distribution of the module size, $Score_{Topo}$ and $Score_{Fun}$ (Supplementary Figure S8). The WGCNA modules displayed very good $Score_{Topo}$, (0.727 ± 0.070) compared to that of RMs (0.147 ± 0.067) and GO-BPs (0.424 ± 0.105) of the same size. Surprisingly, the $Score_{Fun}$ of the WGCNA modules (0.246 ± 0.081) were not very different from those of RMs (0.207 ± 0.032) and

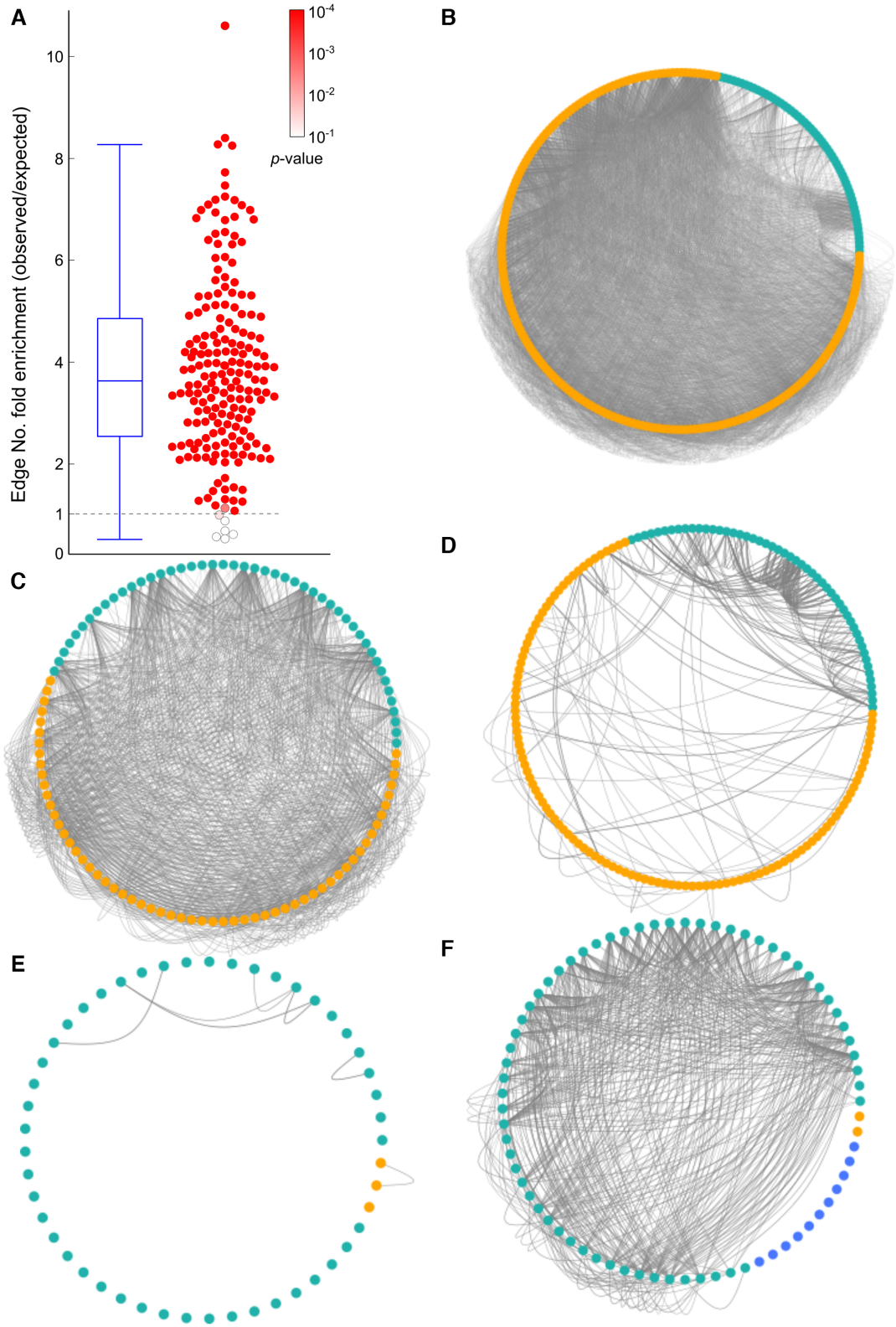

Figure 6. Validation using STRING, a database of known and predicted protein-protein interactions, of the genes gathered by TopoFun in COXPRESdb. (**A**) 187 of the 193 GO-BPs under study displayed a significant enrichment of the number of edges in STRING between the primary genes and the novel genes gathered by TopoFun in COXPRESdb (see Results for details). (**B–F**) Five examples of modules produced by TopoFun and analyzed using STRING. The green dots correspond to the primary genes of the GO-BPs and the orange ones to the novel genes gathered by TopoFun (B) GO:000082~G1/S transition of mitotic cell cycle (edge enrichment, 10.6; *P*-value $< 1 \times 10^{-4}$), (C) GO:0051865~protein autoubiquitination (edge enrichment, 3.90; *P*-value $< 1 \times 10^{-4}$), (D) GO:0043401~steroid hormone mediated signalling pathway (edge enrichment, 1.49; *P*-value $< 1 \times 10^{-4}$), (E) GO:0010923~negative regulation of phosphatase activity (edge enrichment, 0.88; *P*-value, ns), (F) GO:0007596~blood coagulation (edge enrichment, 1.00; *P*-value, ns). The blue dots correspond to primary genes that were part of GO:0010923 and were excluded by TopoFun (see Results for details).

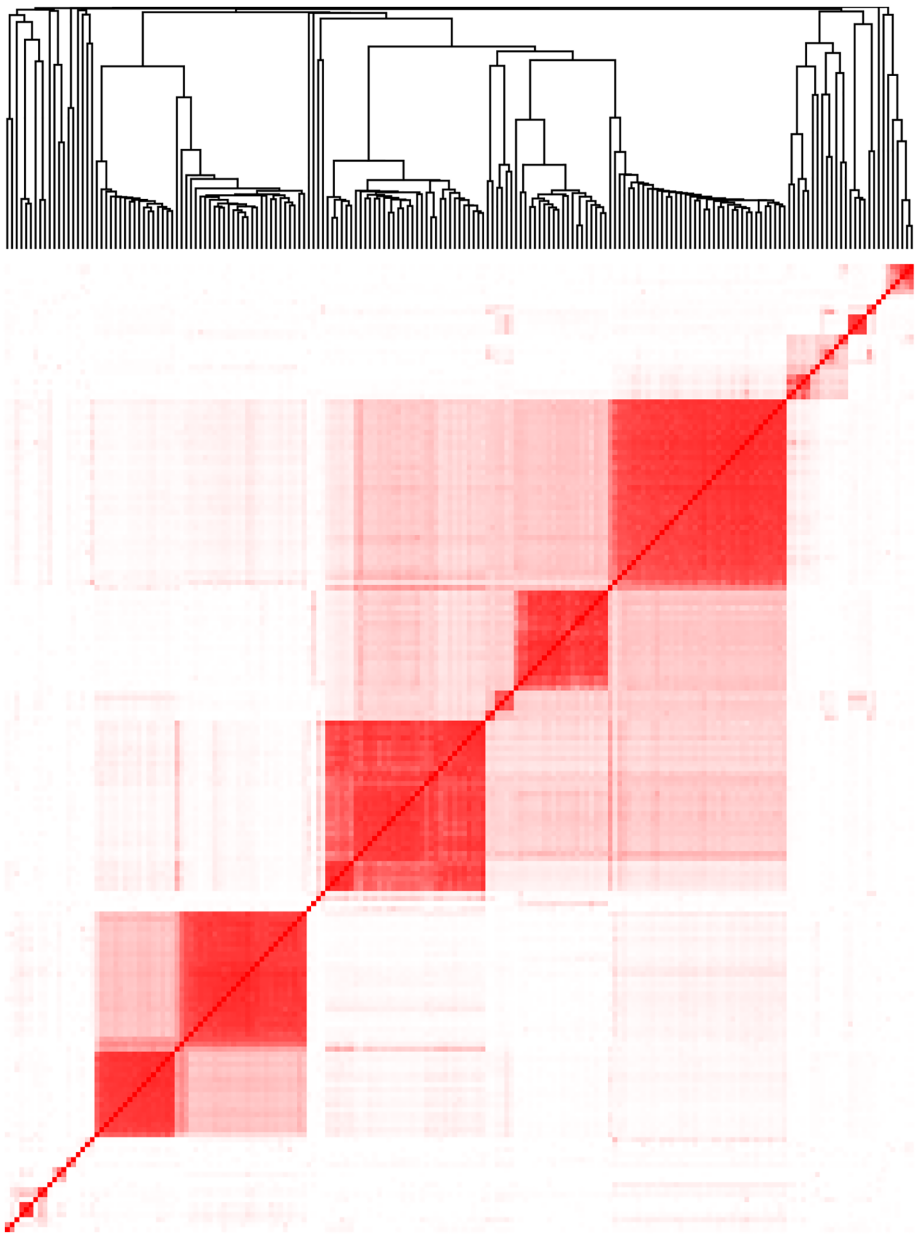

Figure 7. Similarity of the composition of the modules produced by TopoFun. We compared the gene composition of the 'optimized' modules produced by TopoFun from the 193 GO-BPs comprising 50–100 genes. We computed the overlap distance of each pair of modules, performed hierarchical clustering of the 'optimized' modules (top panel), and displayed the results as a heat map (bottom panel). Analysis of the clustering tree supported the existence of 38 clusters, five of which (# 2, 3, 6, 7, 19) comprised numerous modules.

only half of those of GO-BPs (0.480 ± 0.070). After running TopoFun on the WGCNA modules with 50–100 genes, we obtained new modules that displayed a modest (1.09) but significant ($P < 0.01$, paired Student's t-test) increased $Score_{Topo}$ (0.789 ± 0.200). The size of the modules was substantially increased from 70.2 ± 13.7 to 333 ± 32.3, and so were the $Score_{Fun}$ (0.752 ± 0.065). We concluded that the modules produced by WGCNA had an exceptional topology, more compact and dense than most GO-BPs of similar size; however, the constituting genes were only weakly functionally related as their functional similarity was not very different from that of randomly sampled genes. Most often, TopoFun was able to add new genes to these modules without depreciating the module's topology and with a concomitant increase in the functional similarity of the constituting genes.

DISCUSSION

We report on TopoFun, a novel machine learning method aimed at discovering genes co-expressed with and functionally related to a seed gene set. TopoFun required two types of input data: a gene co-expression network (GCN) and a gene annotation set. The choice of both potentially influenced its outcome.

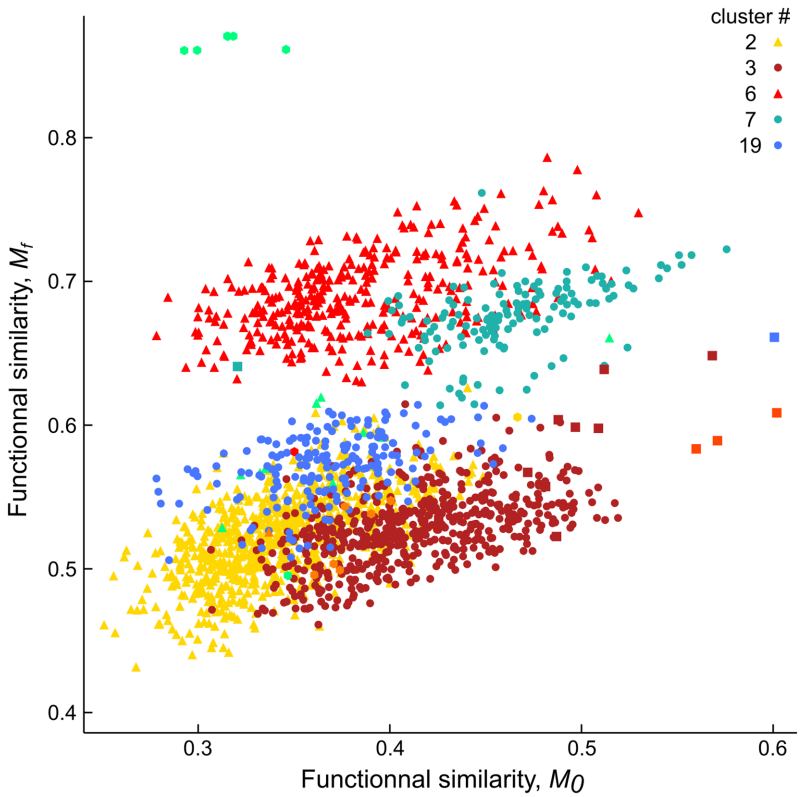

Figure 8. The functional similarity of the initial modules predicted the functional similarity of the modules produced by TopoFun. For any two modules of each cluster displayed in Figure 7, we plotted the functional similarity of the initial modules (M_0) versus that of modules produced by TopoFun (M_f). The five most populated clusters (#2, 38 modules; #3, 34 modules; #6, 27 modules; #7, 17 modules; #19, 20 modules) displayed a significant correlation, indicating that the similarity of the M_f modules is largely dictated by the similarity of the original M_0 modules.

In the present work, the GCN (COXPRESdb) was constructed from 31 479 Affymetrix microarray data. The corresponding biological samples were very diverse and did not represent a specific cell type, tissue or experimental condition. This particular data set prevented the observation of cell type- or tissue-specific co-expression links as all samples were used for the calculation of a single PCC for any gene pair. Despite this limitation, TopoFun was able to aggregate novel genes with existing GO-BPs. A single PCC was also used in GeneFriends, a GCN constructed from RNA-seq data (51). Other co-expression databases used PCC in a way that preserved the information about the samples from which the expression data were obtained. For instance, Gemma recorded the number of independent experiments in which two genes were co-expressed, defined as PCC >0.7 for experiments with 20+ samples (52). This co-expression metrics enables sorting experiments, *e.g.* those derived from a specific tissue, before constructing the GCN; it is then possible to observe sample type-specific co-expression relationships. Another possibility is to construct an *ab initio* GCN from expression data of a specific tissue, cell type or biological condition. The recent availability of single cell expression data in particular offers the opportunity to ask questions about gene co-expression in defined cell types.

TopoFun also required an annotation set, namely Gene Ontology Biological Processes, which greatly influenced the outcome of the method. First, the annotation data determined the construction of the database to train the machine learning model. If some GO-BPs comprised genes that were actually not co-expressed with their BP fellows, one may expect that the topology of some 'functional' modules of the training set was not optimal to identify co-expressed genes. Figures 1 and 3 demonstrated that this situation was not uncommon; a number of functional modules displayed average degree (Supplementary Figure S1) or topological scores (Figures 1 and3) similar to those observed in random modules. One possibility was that some BPs were composed of genes that were not co-regulated; the activity of signalling cascades are often regulated by protein phosphorylation rather than transcriptional co-regulation. The observation of the relative values of $Score_{Topo}$ and $Score_{Fun}$ will help identify gene sets, including GO-BPs, that are transcriptionally co-regulated *vs.* gene sets that are not. Alternatively, some GO-BPs were composed of genes that contributed to the same biological process at the organismal level, but were expressed in different tissues/cell types and should not be called co-expressed. The latter possibility was illustrated by GO:0007596~blood coagulation, as described in the Results section. This example illustrated the ability of TopoFun to discriminate sets of functionally related genes based on co-expression data, even if they were annotated by the same GO-BP.

The above mentioned heterogeneity of some GO-BPs suggests possible improvements of TopoFun. In the present

work we considered the whole GO-BP as a functional module; we could be more restrictive and consider only the largest clique of each GO-BP to train the statistical model. We anticipate that this approach is likely to be too restrictive as only genes co-expressed with every gene in the largest clique are likely to be gathered. A reasonable compromise might be to consider the largest clique and its nearest neighbours to train the model.

The problem of using imperfect functional modules to train the statistical model was also apparent when we measured the classification performance of the topological score. The overall error rate was 7.99%, which is rather high; interestingly, 2.8% of the RMs and 13.4% of the FMs were misclassified (Table 1). The difference in the error rate between RMs and FMs was likely the consequence of the fact that some 'functional' modules consisted of genes that were indeed not co-expressed and correctly classified as RM-like. As a consequence, the FM misclassification rate was likely overestimated. In addition, provided that the RM misclassification rate was low enough, the genetic algorithm could compensate the high FM misclassification rate, as it involved an iterative process, the comparison of numerous modules, and the selection of only the best one. Again, training the model with GO-BP sub-modules rather than the whole GO-BPs should considerably improve the prediction performance of the topological score.

To illustrate TopoFun capabilities, we showed how it assigned novel genes to existing GO-BPs. The novel genes were validated using STRING, an independent knowledge database. We also showed that TopoFun was able to add new genes to modules produced by WGCNA. While preserving the module topology, TopoFun increased the functional similarity of the constituting genes, which was found to be minimal in the original modules. One may argue that, because the aim of network algorithms is precisely to find new modules not previously annotated by a function, TopoFun was too conservative and excessively favoured functionally annotated genes. Because classical network algorithms should also be able to find functionally annotated modules, one may expect that a large proportion of the modules identified by these algorithms would be functionally distinguishable, which was not the case. We concluded that the weighted combination of topological and functional information provided by TopoFun allowed to optimize WGCNA modules to more plausible ones.

The 'optimized' modules produced by TopoFun were generally larger than the original modules they stemmed from (Figure 5). This is a general trend with network algorithms that tend to produce large modules whose biological interpretation is difficult (12,53–55). Heuristic corrections were proposed to remove the bias towards large modules (53,54,56) but resulted in limited efficacy (55). In TopoFun, we payed attention to the problem when we designed the fitness function of the GA. The function was composed of three, equally weighted terms. The third term, $\frac{|M_1 \cap M_0|}{|M_0|}$, was neutral with respect to modules size. It favoured modules that contained the largest proportion of the seed genes; when more genes were added, this term did not increase, unless the added genes were in the initial set of genes (these genes may have been removed at previous iterations). The second term, $Score_{Fun}$, obviously disadvantaged large modules comprising many genes with no functional similarity to the seed genes. $Score_{Fun}$ allowed larger modules only if they consisted of functionally similar genes, preserving the modules interpretability. Finally, only the first term, $\sqrt{\delta_0 + \delta_1 \times Score_{Topo}}$, could favour large modules. Yet, the selected topological descriptors included the size of the modules. This ensured that functional modules of all sizes were compared to random modules of all sizes during the machine learning process. When $Score_{Topo}$ increased following new gene addition, this was mainly because the topological descriptors of the new module were more FM-like, not just because the number of constituting genes increased. TopoFun design was able to minimize the systematic statistical bias uncovered by Nikolayeva and colleagues (12) and produced some large modules that were still interpretable as exemplified by GO:0043401∼ steroid hormone mediated signaling pathway.

Future developments of TopoFun will include the optimization of the training set by incorporating additional sets of functionally related genes. We will test TopoFun on different types of GCNs, in particular cell type-specific GCNs constructed from single cell RNAseq data. We will also use TopoFun for various purposes, *e.g.* the identification of the 'best' functional module of a fixed size including a given gene, the ranking of functional annotations associated with multifunctional genes by including topological information, and, hopefully, the identification of totally new functional modules.

DATA AVAILABILITY

Source code and binaries are available at https://github.com/ljournot/TopoFun/releases. The code is provided as a Jupyter notebook that runs in the Jupyter/Jupyterlab IDE with an R kernel. The documentation is available in the notebook, which also includes an application to GO:0006413∼translational initiation.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

ACKNOWLEDGEMENTS

The authors wish to thank Drs T. Bouschet and A. Varrault for critical reading of the manuscript. They wish to express their gratitude to the two anonymous reviewers who helped improve the manuscript by their detailed, in-depth and constructive thoughts and questions about TopoFun merits.

FUNDING

Centre National de la Recherche Scientifique; Institut National de la Santé et de la Recherche Médicale; Université de Montpellier; A.J. was supported by the Université de Montpellier; Centre National de la Recherche Scientifique-Lebanon; Université Libanaise; M.G.X. acknowledges the financial support from France Génomique National Infrastructure, funded as part of the 'Investissement d'Avenir' program managed by the Agence Nationale pour la Recherche [ANR-10-INBS-09]. Funding for open access charge: Centre National de la Recherche Scientifique.

Conflict of interest statement. None declared.

REFERENCES

1. Stoeger,T., Gerlach,M., Morimoto,R.I. and Nunes Amaral,L.A. (2018) Large-scale investigation of the reasons why potentially important genes are ignored. *PLoS Biol.*, **16**, e2006643.
2. The Gene Ontology Consortium (2017) Expansion of the gene ontology knowledgebase and resources. *Nucleic Acids Res.*, **45**, D331–D338.
3. Kanehisa,M., Goto,S., Sato,Y., Furumichi,M. and Tanabe,M. (2012) KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.*, **40**, D109–D114.
4. Jassal,B., Matthews,L., Viteri,G., Gong,C., Lorente,P., Fabregat,A., Sidiropoulos,K., Cook,J., Gillespie,M., Haw,R. *et al.* (2020) The reactome pathway knowledgebase. *Nucleic Acids Res.*, **48**, D498–D503.
5. Stuart,J.M., Segal,E., Koller,D. and Kim,S.K. (2003) A gene-coexpression network for global discovery of conserved genetic modules. *Science*, **302**, 249–255.
6. van Noort,V., Snel,B. and Huynen,M.A. (2003) Predicting gene function by conserved co-expression. *Trends Genet.*, **19**, 238–242.
7. Hansen,B.O., Vaid,N., Musialak-Lange,M., Janowski,M. and Mutwil,M. (2014) Elucidating gene function and function evolution through comparison of co-expression networks of plants. *Front Plant Sci*, **5**, 394.
8. Schaefer,R.J., Michno,J.-M. and Myers,C.L. (2017) Unraveling gene function in agricultural species using gene co-expression networks. *Biochim. Biophys. Acta*, **1860**, 53–63.
9. Liesecke,F., Daudu,D., Dugé de Bernonville,R., Besseau,S., Clastre,M., Courdavault,V., de Craene,J.-O., Crèche,J., Giglioli-Guivarc'h,N., Glévarec,G. *et al.* (2018) Ranking genome-wide correlation measurements improves microarray and RNA-seq based global and targeted co-expression networks. *Sci. Rep.*, **8**, 10885.
10. Mitra,K., Carvunis,A.-R., Ramesh,S.K. and Ideker,T. (2013) Integrative approaches for finding modular structure in biological networks. *Nat. Rev. Genet.*, **14**, 719–732.
11. Fortunato,S. and Hric,D. (2016) Community detection in networks: a user guide. *Phys. Rep.*, **659**, 1–44.
12. Nikolayeva,I., Guitart Pla,O. and Schwikowski,B. (2018) Network module identification—a widespread theoretical bias and best practices. *Methods*, **132**, 19–25.
13. Langfelder,P. and Horvath,S. (2008) WGCNA: an r package for weighted correlation network analysis. *BMC Bioinformatics*, **9**, 559.
14. Saelens,W., Cannoodt,R. and Saeys,Y. (2018) A comprehensive evaluation of module detection methods for gene expression data. *Nat. Commun.*, **9**, 1090.
15. Hric,D., Darst,R.K. and Fortunato,S. (2014) Community detection in networks: structural communities versus ground truth. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, **90**, 062805.
16. Peel,L., Larremore,D.B. and Clauset,A. (2017) The ground truth about metadata and community detection in networks. *Sci. Adv.*, **3**, e1602548.
17. Nguyen,H., Shrestha,S., Tran,D., Shafi,A., Draghici,S. and Nguyen,T. (2019) A comprehensive survey of tools and software for active subnetwork identification. *Frontiers in Genetics*, **10**, 155.
18. Choobdar,S., Ahsen,M.E., Crawford,J., Tomasoni,M., Fang,T., Lamparter,D., Lin,J., Hescott,B., Hu,X., Mercer,J. *et al.* (2019) Assessment of network module identification across complex diseases. *Nat. Methods*, **16**, 843–852.
19. Sharan,R., Maron-Katz,A. and Shamir,R. (2003) CLICK and EXPANDER: a system for clustering and visualizing gene expression data. *Bioinformatics*, **19**, 1787–1799.
20. Ponzoni,I., Nueda,M., Tarazona,S., Götz,S., Montaner,D., Dussaut,J., Dopazo,J. and Conesa,A. (2014) Pathway network inference from gene expression data. *BMC Syst. Biol.*, **8**(Suppl. 2), S7.
21. Leale,G., Baya,A.E., Milone,D.H., Granitto,P.M. and Stegmayer,G. (2018) Inferring unknown biological function by integration of GO annotations and gene expression data. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **15**, 168–180.
22. Ballouz,S., Weber,M., Pavlidis,P. and Gillis,J. (2017) EGAD: ultra-fast functional analysis of gene networks. *Bioinformatics*, **33**, 612–614.
23. Wang,Y., Yang,S., Zhao,J., Du,W., Liang,Y., Wang,C., Zhou,F., Tian,Y. and Ma,Q. (2019) Using machine learning to measure relatedness between genes: a multi-features model. *Sci. Rep.*, **9**, 4192.
24. Botía,J.A., Vandrovcova,J., Forabosco,P., Guelfi,S., D'Sa,K. and United Kingdom Brain Expression ConsortiumUnited Kingdom Brain Expression Consortium, Hardy,J., Lewis,C.M., Ryten,M. and Weale,M.E. (2017) An additional k-means clustering step improves the biological features of WGCNA gene co-expression networks. *BMC Syst. Biol.*, **11**, 47.
25. Margolin,A.A., Nemenman,I., Basso,K., Wiggins,C., Stolovitzky,G., Dalla Favera,R. and Califano,A. (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, **7**(Suppl. 1), S7.
26. Franz,M., Rodriguez,H., Lopes,C., Zuberi,K., Montojo,J., Bader,G.D. and Morris,Q. (2018) GeneMANIA update 2018. *Nucleic Acids Res.*, **46**, W60–W64.
27. Hartwell,L.H., Hopfield,J.J., Leibler,S. and Murray,A.W. (1999) From molecular to modular cell biology. *Nature*, **402**, C47–C52.
28. Al Adhami,H., Evano,B., Le Digarcher,A., Gueydan,C., Dubois,E., Parrinello,H., Dantec,C., Bouschet,T., Varrault,A. and Journot,L. (2015) A systems-level approach to parental genomic imprinting: the imprinted gene network includes extracellular matrix genes and regulates cell cycle exit and differentiation. *Genome Res.*, **25**, 353–367.
29. Tibshirani,R. (1996) Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. B (Methodological)*, **58**, 267–288.
30. Fisher,R.A. (1936) The use of multiple measurements in taxonomic problems. *Ann. Eugen.*, **7**, 179–188.
31. Wang,J.Z., Du,Z., Payattakool,R., Yu,P.S. and Chen,C.-F. (2007) A new method to measure the semantic similarity of GO terms. *Bioinformatics*, **23**, 1274–1281.
32. Obayashi,T., Okamura,Y., Ito,S., Tadaka,S., Motoike,I.N. and Kinoshita,K. (2013) COXPRESdb: a database of comparative gene coexpression networks of eleven species for mammals. *Nucleic Acids Res.*, **41**, D1014–D1020.
33. Obayashi,T. and Kinoshita,K. (2009) Rank of correlation coefficient as a comparable measure for biological significance of gene coexpression. *DNA Res.*, **16**, 249–260.
34. Maslov,S. and Sneppen,K. (2002) Specificity and stability in topology of protein networks. *Science*, **296**, 910–913.
35. Dong,J. and Horvath,S. (2007) Understanding network concepts in modules. *BMC Syst. Biol.*, **1**, 24.
36. Assenov,Y., Ramírez,F., Schelhorn,S.-E., Lengauer,T. and Albrecht,M. (2008) Computing topological parameters of biological networks. *Bioinformatics*, **24**, 282–284.
37. Hastie,T., Tibshirani,R. and Friedman,J. (2001) *The Elements of Statistical Learning*. Springer New York Inc., New York, NY, USA.
38. Wheeler,D.L., Barrett,T., Benson,D.A., Bryant,S.H., Canese,K., Chetvernin,V., Church,D.M., Dicuccio,M., Edgar,R., Federhen,S. *et al.* (2008) Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, **36**, D13–D21.
39. Jiang,J.J. and Conrath,D.W. (1997) Semantic similarity based on corpus statistics and lexical taxonomy. In: *Proceedings of the 10th Research on Computational Linguistics International Conference*. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Taipei, Taiwan, pp. 19–33.
40. Lin,D. (1998) An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning, ICML '98*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 296–304.
41. Resnik,P. (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.*, **11**, 95–130.
42. Goldberg,D.E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning 1st ed*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
43. Chandrashekar,G. and Sahin,F. (2014) A survey on feature selection methods. *Comput. Electr. Eng.*, **40**, 16–28.
44. Soufan,O., Kleftogiannis,D., Kalnis,P. and Bajic,V.B. (2015) DWFS: a wrapper feature selection tool based on a parallel genetic algorithm. *PLoS One*, **10**, e0117988.
45. Reeves,C.R. and Rowe,J.E. (2002) *Genetic Algorithms: Principles and Perspectives. A Guide to GA Theory*. Springer, US.

46. Bhandari,D., Murthy,C.A. and Pal,S.K. (1996) Genetic algorithm with elitist model and its convergence. *Int. J. Patt. Recogn. Artif. Intell.*, **10**, 731–747.
47. Wang,H., Azuaje,F., Bodenreider,O. and Dopazo,J. (2004) Gene expression correlation and gene ontology-based similarity: an assessment of quantitative relationships. *Proc IEEE Symp. Comput. Intell. Bioinforma Comput. Biol.*, **2004**, 25–31.
48. Wolfe,C.J., Kohane,I.S. and Butte,A.J. (2005) Systematic survey reveals general applicability of 'guilt-by-association' within gene coexpression networks. *BMC Bioinformatics*, **6**, 227.
49. Szklarczyk,D., Gable,A.L., Nastou,K.C., Lyon,D., Kirsch,R., Pyysalo,S., Doncheva,N.T., Legeay,M., Fang,T., Bork,P. *et al.* (2021) The STRING database in 2021: customizable protein-protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Res.*, **49**, D605–D612.
50. Hochberg,Y. and Benjamini,Y. (1990) More powerful procedures for multiple significance testing. *Stat. Med.*, **9**, 811–818.
51. van Dam,S., Craig,T. and de Magalhães,J.P. (2015) GeneFriends: a human RNA-seq-based gene and transcript co-expression database. *Nucleic Acids Res.*, **43**, D1124–D1132.
52. Zoubarev,A., Hamer,K.M., Keshav,K.D., McCarthy,E.L., Santos,J.R.C., Van Rossum,T., McDonald,C., Hall,A., Wan,X., Lim,R. *et al.* (2012) Gemma: a resource for the reuse, sharing and meta-analysis of expression profiling data. *Bioinformatics*, **28**, 2272–2273.
53. Rajagopalan,D. and Agarwal,P. (2005) Inferring pathways from gene lists using a literature-derived network of biological relationships. *Bioinformatics*, **21**, 788–793.
54. Nacu,S., Critchley-Thorne,R., Lee,P. and Holmes,S. (2007) Gene expression network analysis and applications to immunology. *Bioinformatics*, **23**, 850–858.
55. Batra,R., Alcaraz,N., Gitzhofer,K., Pauling,J., Ditzel,H.J., Hellmuth,M., Baumbach,J. and List,M. (2017) On the performance of de novo pathway enrichment. *NPJ Syst. Biol. Appl.*, **3**, 6.
56. Liu,Y., Brossard,M., Roqueiro,D., Margaritte-Jeannin,P., Sarnowski,C., Bouzigon,E. and Demenais,F. (2017) SigMod: an exact and efficient method to identify a strongly interconnected disease-associated module in a gene network. *Bioinformatics*, **33**, 1536–1544.