PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE

SCHOOL OF ENGINEERING

A MACHINE LEARNING APPROACH TO PREDICT GENE EXPRESSION SIGNATURES, LOCAL GENE NETWORKS, AND KEY GENES FOR BIOLOGICAL FUNCTIONS OF INTEREST

TOMAS FRANCISCO PUELMA PETERS

Thesis submitted to the Office of Graduate Studies in partial fulfillment of the requirements for the Degree of Doctor in Engineering Sciences.

Advisors:
ALVARO SOTO
RODRIGO A. GUTIERREZ

Santiago de Chile, October, 2015

PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE

SCHOOL OF ENGINEERING

A MACHINE LEARNING APPROACH TO PREDICT GENE EXPRESSION SIGNATURES, LOCAL GENE NETWORKS, AND KEY GENES FOR BIOLOGICAL FUNCTIONS OF INTEREST

TOMAS FRANCISCO PUELMA PETERS

Members of the Committee:

ALVARO SOTO

RODRIGO A. GUTIERREZ

ALEJANDRO MAASS

FRANCISCO MELO

KARIM PICHARA

LAURENCE LEJAY

CRISTIÁN VIAL

Thesis submitted to the Office of Graduate Studies in partial fulfillment of the requirements for the Degree of Doctor in Engineering Sciences.

Santiago de Chile, October, 2015

A mis padres, familiares y amigos,
que me apoyaron.

AGRADECIMIENTOS

Quiero agradecer a mi familia y de forma especial a mi madre, Karin Peters, por su apoyo incondicional en este proceso de estudio. A mis profesores, Alvaro Soto y Rodrigo Gutierrez, que con su guía me enseñaron cosas que no pueden ser aprendidas en libros y que serán claves en mi vida personal y profesional. Finalmente, a mi abuelo, Francisco Puelma (q.e.p.), que me transmitió desde niño la pasión por la ingeniería.

INDEX

INDEX OF TABLES

INDEX OF FIGURES

ABSTRACT

A major aim of biological research is to discover relevant genes for biological functions of interest. These functions may have different levels of complexity, from specific biological processes, to complex traits such as "Metabolic diseases" in humans or "Flowering time" in plants. Unfortunately, the large number of genes, and the many and intricate interactions among them, make difficult for biologist to discern which genes to study.

This thesis present Discriminative Local Subspaces (DLS), a method that combines supervised machine learning and coexpression techniques to predict gene networks that can pinpoint new and key genes for specific biological functions of interest. It also presents GENIUS, a web server with a user-friendly interface for DLS that allows the scientific community to fully exploit its capabilities. Unlike traditional coexpression networks (CNs), DLS uses the knowledge available in Gene Ontology (GO) to generate informative training sets that guide the discovery of expression signatures: expression patterns that are discriminative for genes involved in the biological function of interest. By linking genes coexpressed with these signatures, DLS is able to construct a gene network that links both, known and new genes, for the biological function of interest.

Our systematic evaluations demonstrate DLS can predict new genes with accuracies comparable to highly discriminative Support Vector Machine (SVM) methods, while maintaining the informative and useful representation of CNs. Moreover, they show that unlike SVMs and CNs, DLS can systematically improve its prediction performance as more experimental data becomes available. Remarkably, our evaluations in real research scenarios show that GENIUS can be effectively used to make novel discoveries. In particular, GENIUS predicted a novel and key gene to improve nitrogen use efficiency of plants, which was experimentally validated. Therefore, we believe GENIUS can aid biologists to generate concrete hypothesis from prior knowledge, make novel discoveries, and ultimately, improve our molecular understanding of biological systems. GENIUS currently supports eight mayor organisms and is freely available for public use at http://networks.bio.puc.cl/genius.

Keywords: supervised machine learning, local gene networks, coexpression networks, gene function prediction, expression signatures, web application, bioinformatics.

RESUMEN

Un objetivo importante en la investigación biológica es descubrir genes relevantes para funciones biológicas de interés. Estas funciones pueden tener distintos grados de complejidad, desde procesos biológicos específicos a características complejas como "enfermedades metabólicas" en humanos o "tiempo de floración" en plantas.

Esta tesis presenta "Discriminative Local Subspaces" (DLS), un método computacional que combina técnicas de aprendizaje de máquina supervisado y coexpresión para predecir redes de genes capaces de exponer nuevos genes y señalar cuáles son importantes para una función biológica de interés. Además presenta GENIUS, un servidor web que permite a la comunidad científica aprovechar las capacidades de DLS a través de una interfaz amistosa. En contraste con las redes de coexpresión tradicionales, DLS usa el conocimiento disponible en Gene Ontology (GO) para generar conjuntos de entrenamiento informativos y guiar la búsqueda de "firmas de expresión": patrones de expresión distintivos para los genes que participan en una función biológica de interés. Luego, DLS une los genes que se coexpresan con estas firmas para formar una red génica que contiene tanto genes conocidos como nuevos, para la función biológica de interés.

Nuestras evaluaciones sistemáticas demuestran que DLS puede predecir genes nuevos con precisiones comparables a los métodos altamente discriminativos "Support Vector Maquines" (SVM), pero manteniendo la representación intuitiva e informativa que proveen las redes de coexpresión. Más aún, nuestras evaluaciones muestran que, en contraste con estos métodos, DLS mejora su precisión sistemáticamente al aumentar la cantidad de datos experimentales disponibles. Además, nuestras evaluaciones en escenarios reales muestran que GENIUS puede realizar nuevos descubrimientos de manera efectiva. En particular, GENIUS predijo un nuevo gen, clave para mejorar la eficiencia del uso del nitrógeno en plantas, lo cual fue validado experimentalmente. Por lo tanto, creemos que GENIUS puede ayudar a generar hipótesis concretas a partir de conocimiento previo, realizar nuevos descubrimientos y, por lo tanto, mejorar nuestra comprensión molecular sobre los sistemas biológicos. GENIUS actualmente puede ser utilizado en ocho importantes organismos y es de libre acceso en http://networks.bio.puc.cl/genius.

1. INTRODUCTION

All living organisms have a genome; a library of genes that define the potential characteristics they can develop. All cells within an organism have access to an exact copy of their genome, which is contained in a large molecule called DNA. Making an analogy between a cell and a computer, genes can be thought as the source code of proteins, which in turn, are the programs that give a cell its unique traits. In the same way that source codes are compiled into programs by a compiler, which is yet another program, genes are converted into proteins by other proteins. This conversion occurs in a two steps process. First, a gene is transcribed from the DNA molecule to a temporal RNA molecule that contains the code of only this transcribed gene. Then, this RNA molecule is translated into a protein. The amount of RNA available of a given gene in a given moment is called the expression level of this gene, and it can be used as an approximation of the amount of protein available that is codified by that gene. Gene expression level can also be induced or repressed by other proteins, which are called transcription factors. In addition, there are proteins that can sense internal and external cell conditions, and communicate this information to transcription factors, which then adapt the expression level of genes in order to cope for the sensed environmental changes. Thus, although the genome defines the spectrum of potential genes that can be expressed in an organism, the internal and external conditions of a specific cell define the precise genes that are expressed in a given moment, and thus, the precise characteristics that it will display.

In the last decade, the complete genomes of hundreds of organisms have been sequenced and thousands of genes have been identified. Despite these efforts, biology still faces the relevant challenge of discovering the biological functions and traits in which many of these genes are involved inside cells. For example, *Arabidopsis thaliana*, the model organism in plant molecular biology, has 16,319 (52%) genes that are not annotated in any biological process of the Gene Ontology (GO) project database (Ashburner et al., 2000). Moreover, the large number of genes, and the many and intricate interactions among them, makes difficult for biologist to discern which genes are most relevant for a specific biological function of interest. These functions may have different levels of complexity, from specific biological processes such as “Response to heat”, to complex traits that involve several interacting processes such as “Metabolic diseases” in humans or “Flowering time” in plants. As a consequence, biologists use a mix of prior knowledge and intuition to choose which genes to focus and which to ignore for detailed experimental work (Moreau & Tranchevent, 2012). This often leads to high costs in both time and experimental research. Discovering which genes are important for a biological function of interest is key to understand the underlying molecular mechanisms that govern traits of interest, which can lead to important biotechnological applications.

By using high-throughput technologies, such as microarrays, scientists have now the opportunity to obtain large amounts of genomic data in a single experiment. A single microarray experiment allows scientists to measure the expression level of almost the entire genome of an organism, providing a snapshot of its state under a given experimental condition. Moreover, many databases offer open access to the data of thousands of microarray experiments for many organism, which opens new opportunities to analyze the transcriptional and functional relationships among genes. However, manual analysis of these large datasets is not a viable solution.

Given the problems stated above, the primary goal of this thesis is to develop a computational approach to enable biologist to extract feasible biological hypothesis from large-scale gene expression datasets. Consequently, this thesis has the following specific goals:

1. Develop a novel algorithm to predict new and key genes related to biological functions of interest.
2. Make a biological contribution by applying the developed algorithm to a real research scenario.
3. Develop a user-friendly interface for the algorithm to allow the scientific community to fully exploit its capabilities.

In order to achieve the goals presented above, this thesis presents a novel supervised machine learning method, called Discriminative Local Subspaces (DLS), designed specifically to analyze gene expression data and to extract relevant information that biologists can use to guide research. Unlike other methods, DLS is designed to predict local and discriminative gene networks. DLS uses semantic information available in GO annotations and machine learning techniques to focus the inference of networks on “local subspaces” of the experimental data. These local subspaces are also “discriminative”, because they are defined by “expression signatures”: expression patterns found in particular genes and experimental conditions that are distinctive of the genes of the biological function of interest defined by the user. In addition, DLS uses a supervised-learning approach to infer gene associations, which allows it to predict new genes related to the biological function of interest with accuracies comparable to highly discriminative Support Vector Machine (SVM) methods (Puelma, Gutierrez, & Soto, 2012). Moreover, and unlike SVMs, DLS maintains the simplicity and intuitive representation of coexpression networks and systematically improves its prediction performance as more experimental conditions are added to a dataset (Puelma et al., 2012).

In order to allow biologists to fully exploit the capabilities of DLS and guide their researches, this thesis also presents GENIUS (GEne Networks Inference Using Signatures), a web server with a user-friendly interface to DLS. GENIUS incorporates Gene Ontology annotations and thousands of microarrays experiments from Gene Expression Omnibus (GEO) for eight model organisms: *Arabidopsis thaliana, Caenorhabditis elegans, Danio rerio, Drosophila melanogaster, Escherichia coli, Homo sapiens, Mus musculus,* and *Saccharomyces cerevisiae*. In addition, GENIUS provides several tools to visualize and analyze the gene networks predicted by DLS, including integration with Cytoscape (Smoot, Ono, Ruscheinski, Wang, & Ideker, 2011), an advanced network analysis software platform. Finally, GENIUS offers simple scores and graph theory indicators to pinpoint key genes in the predicted networks, making it easy for biologist to prioritize genes for functional validation assays. These salient features make GENIUS a powerful tool to develop concrete hypothesis and to identify relevant genes for biological functions of interest. GENIUS is freely accessible at http://networks.bio.puc.cl/genius.

The rest of this thesis is organized as follows. Chapter 2 reviews related state of the art methods and software, highlighting the contributions of DLS and GENIUS to overcome their limitations. Chapter 3 presents the theoretical bases behind DLS, describes the server architecture of GENIUS, the datasets and supported organisms, and the scores and graph theory indicators incorporated for gene prioritization. Chapter 4 presents an extensive and systematic evaluation that compares the performance of DLS against Support Vector Machines (Brown et al., 2000) and Coexpression Networks (Vandepoele et al., 2009), using datasets from years 2008 and 2010. Then, it shows an updated evaluation using a bigger dataset from year 2015 for the eight organism currently supported by GENIUS. In addition, it presents three case studies that show the practical usefulness of GENIUS to guide research. Finally, chapter 5 summarizes the main conclusions of this thesis and presents some future avenues of research.

2. RELATED WORK

Machine learning (Mitchell, 1997) has emerged as an important technology to support gene function discovery. In particular, many methods have been proposed to take advantage of the massive amounts of microarray expression data available (see Valafar, 2002; Zhao et al., 2008 for reviews). However, as this chapter shows, these methods have not fully exploited the potential of both, machine learning and microarray data. Among these methods, we can identify three main groups that allow us to illustrate the main limitations of current techniques, as well as, to highlight the main advantages of the proposed DLS method. These three groups correspond to: i) supervised machine learning methods, ii) coexpression based methods, and iii) biclustering methods. This chapter starts reviewing each of these groups. Afterwards, it reviews relevant public web applications to infer gene networks and gene functions, comparing them with GENIUS.

2.1. Supervised methods

Supervised machine learning consists in programming computer methods to optimize predictive models by using training data or past experience (Larranaga et al., 2006). These models can be optimized to predict either discrete or continuous data. However, in this thesis we are interested in the discrete case, where methods are used to classify data samples in discrete classes or labels. These methods are composed by two main processes: training and classification. During training, they use a set of samples that have been labeled in different classes to learn patterns, or mathematical functions, that can then be used to predict the class label of new samples. In this thesis, a sample corresponds to the expression profile of a particular gene, i.e., a vector containing the expression level of a gene in different experimental conditions, and the labels, or categories, correspond to the biological function in which a given gene is involved.

Most supervised machine learning methods that have been used successfully for gene function prediction are based on discriminative black-box schemes (Mitchell, 1997) and, in particular, maximum margin classifiers such as a Support Vector Machines (SVMs) (Cortes & Vapnik, 1995). Brown et al. (2000) were one of the first researchers to evaluate the potential of supervised methods in expression data to predict gene function. They tested five methods for the classification of yeast genes in six functional classes and concluded that SVMs outperform the prediction results of other four commonly used machine learning algorithms: Fisher linear discriminant, Parzen windows, and two flavors of decision tree classifiers. Later, Mateos et al. (2002) compared the performance of Artificial Neural Networks (ANNs) with SVMs, using the same dataset reported by Brown et al. (2000). Their results indicated that ANNs perform comparably, but slightly worse than SVMs in terms of precision. Furthermore, they did a deeper evaluation using 96 functional classes, showing that only 8 of them could be learned with recalls greater that 40%. Thus, among supervised machine learning techniques, SVMs have been one of the most successful approaches to predict gene function (Barutcuoglu, Schapire, & Troyanskaya, 2006; Brown et al., 2000; Mateos et al., 2002; Yang, 2004). However, the transparency and interpretability of a predictive model can be as important as their prediction accuracy (Larranaga et al., 2006). Despite the theoretical advantage of SVMs in terms of classification accuracy, in practice, they present the mayor inconvenience of operating as a black-box (Barakat & Bradley, 2010). Although additional techniques can be applied to extract comprehensible semantic information from SVM models, their application is not straightforward and is usually restricted to linear-SVM models (Fung, Sandilya, & Rao, 2005; Guyon, Weston, Barnhill, & Vapnik, 2002; Wang, Han, & Yan, 2009). In the general case of non-linear SVMs, the transformation of the data to high-dimensional spaces complicates any interpretation of the SVM solution. In our experience, this is a major limitation of SVMs for gene function discovery, as understanding the predictions is a key aspect to evaluate their biological soundness and to guide research. This aspect is even more critical considering the incomplete nature of gene functional annotations and the capability of genes to have multiple functions, which prevent to obtain an error-free gold standard, and thus to evaluate the absolute accuracy of the methods (false-negative problem) (Jansen & Gerstein, 2004; Mateos et al., 2002).

2.2. Semi-supervised methods

In response to the limitations of black-box methods, a second group of more informative methods to predict gene functions have been proposed (Bassel et al., 2011; Eisen, Spellman, Brown, & Botstein, 1998; Horan et al., 2008; S. K. Kim et al., 2001; I. Lee, Ambaru, Thakkar, Marcotte, & Rhee, 2010; Stuart, Segal, Koller, & Kim, 2003; Vandepoele et al., 2009). These methods are based in semi-supervised approaches, which first group genes in an unsupervised manner, without using any functional information, and then a prediction is performed, usually by propagating the over-represented functions among the genes of each group (‘guilt-by-association’ rule) (Walker, Volkmuth, Sprinzak, Hodgson, & Klingler, 1999). The basic assumption in these methods is that if a group of genes show synchronized (correlated) expression patterns along many experimental conditions, then there is a high chance for they to participate in a common biological function. Common techniques used to group genes are clustering (Alon et al., 1999; Eisen et al., 1998; Horan et al., 2008), biclustering (Madeira & Oliveira, 2004; Prelić et al., 2006; Tanay, Sharan, & Shamir, 2005), and coexpression networks (CNs) (Bassel et al., 2011; Stuart et al., 2003; Vandepoele et al., 2009). In particular, CNs are one of the most extensively used tools in the Systems Biology field. This is mainly due to their rich representation, which usually provides biologically meaningful concepts that help scientists to obtain insights about the predictions. Briefly, a CN consists of a graph that is made by connecting genes whose expression patterns show a correlation greater than a given threshold. Then, a functional term is predicted for a gene if it is statistically overrepresented among its neighbors in the network (Vandepoele et al., 2009).

Unfortunately, coexpression based methods have several inconveniences. In particular, the predictions of these methods are in general less accurate than the outputs of supervised methods, as we show in chapter 4 by comparing the performances of CNs, SVMs, and DLS. Furthermore, they typically need a suitable correlation threshold selected by the user to define coexpressed genes, which is often difficult and arbitrary (Vandepoele et al., 2009). In addition, a major drawback of both, CNs and clustering methods, is that they rely on global coexpression patterns, meaning that genes need to be coexpressed in a large proportion of the data in order to be grouped together. Usually, this data involves thousands of microarray experiments, each measured under a wide range of conditions, such as different time points, tissues, environmental conditions, genetic backgrounds, and mutations. Based on general understanding of cellular processes, genes are usually coregulated, and thus coexpressed, only under certain experimental conditions. As a consequence, expecting for genes to be coexpressed in all the available conditions becomes a very strong imposition and limitation.

The previous observation has motivated the development of biclustering methods. The main idea behind these methods is to find subsets of genes coexpressed in subsets of experimental conditions. After the seminal work by Cheng and Church (2000), an extensive list of biclustering approaches has been proposed (Madeira & Oliveira, 2004; Prelić et al., 2006; Tanay et al., 2005). However, besides their theoretical advantages, these approaches have not been extensively used in practice. Based on our own experience, a mayor limitation of these techniques is their unsupervised approach to search for local coexpression patterns, which means that they do not use the known labels or functions of genes to search these patterns. In particular, this blind search can lead to clusters where genes of a broad range of functions are coexpressed, thus, limiting their discriminative properties. This problem is even worse when considering the noisy nature of microarray data, which often leads to the discovery of small and meaningless biclusters that obscure the finding of relevant associations. Selecting datasets in a 'condition-dependent' fashion should more precisely identify gene interactions relevant to a specific biological question at hand (Bassel et al., 2011). However, given the amount of expression data available today, manual selection of relevant experimental conditions is not a practical solution in most cases.

2.3. Our Approach: Discriminative Local Subspaces (DLS)

DLS overcomes the state-of-the-art limitations exposed above by taking advantage of the discriminative nature of supervised machine learning methods, while at the same time, maintaining the expressiveness of coexpression networks approaches.

Unlike semi-supervised coexpression-based approaches, like biclustering, DLS exploits the existing knowledge available in Gene Ontology to construct informative training sets and guide the search of suitable subsets of experimental conditions. These subsets of experimental conditions are searched in the form of expression signatures. An expression signature corresponds to a discriminative expression pattern with two key properties: i) it is defined by a particular gene and a subset of experimental conditions (i.e. it is relevant to a local subspace of the data) and (ii) it is highly discriminative, and therefore useful to identify positive genes associated to a particular biological function of interest (Figure 3.1). The discriminative nature of expression signatures allows DLS to reveal novel coexpression associations for the selected biological function. As a further feature, and to tackle the inherent noise of negative training sets (genes not related to a biological process), DLS incorporates a procedure that iteratively predicts false-negative genes and refines the training set in order to improve its prediction performance.

In contrast to discriminative black-box models, such as SVM, DLS exposes the predicted associations in the context of a discriminative coexpression network, giving the scientist the possibility to visualize, evaluate, and interpret them. However, unlike traditional CN, DLS does not rely on a predefined and fixed correlation threshold to infer the networks. Instead, DLS uses a Bayesian probabilistic approach that adaptively derives a confidence score for each predicted association. A network is then constructed based on a desired minimum confidence, which is translated into different correlation thresholds depending on the discriminative level of each signature.

As shown in chapter 4, our results reveal that DLS attains superior average accuracy and similar predictive power than radial-basis SVM. Furthermore, they show a clear advantage to DLS over linear SVM and CN. Remarkably, they show that unlike SVM and CN, DLS is able to systematically improve its predictive power when increasing the number of available experimental conditions.

2.4 Online tools to predict gene networks and genes functions

There are several tools available in the web to generate and analyze gene networks (Franceschini et al., 2013; Jupiter, Chen, & VanBuren, 2009; Kao & Gunsalus, 2008; Katari et al., 2010; Moreau & Tranchevent, 2012; Obayashi et al., 2013; Obayashi, Nishida, Kasahara, & Kinoshita, 2011; Zuberi et al., 2013). Despite the limitations of current coexpression based approaches, in practice, biologists prefer these more informative methods over black-box supervised methods. In particular, coexpression networks have become the most popular approach to analyze gene expression data and predict gene function. They provide an intuitive graphical representation and meaningful biological insights that help to understand and evaluate the quality of the predictions.

For example, GeneMANIA (Zuberi et al., 2013) and STRING (Franceschini et al., 2013) are popular state-of-the-art web tools that include vast databases for several organisms. They are able to generate gene networks starting from a query list of genes of interest. These tools are powered by fast algorithms and are able to combine multiple kinds of interactions in a unique network representation. These interactions include gene coexpression, protein colocalization, genetic interactions, and shared protein domains. In particular, GeneMANIA uses a fast network weighting algorithm to generate a composite network, by combining many precomputed networks coming from these different sources. Then, it uses a label propagation algorithm to score each gene. Finally, it selects the subnetwork that contains the genes in the query list and 20 additional genes with the highest scores according to the label propagation algorithm (Zuberi et al., 2013).

There are three key aspects that distinguish GENIUS from these state of the art tools. i) Users without bioinformatics skills can guide predictions using prior knowledge, starting from a set of Gene Ontology functional terms, besides a predefined set of genes. ii) Predictions are local and discriminative, which allows GENIUS to uncover relevant coexpression relationships that other tools would miss, while, at the same time, filtering out relations that are trivial or not related to the biological function of interest. iii) GENIUS offers simple scores and graph theory indicators to pinpoint key genes in the predicted networks, making it easier for biologists to prioritize genes for functional validation assays.

These salient features make GENIUS a powerful tool to develop concrete hypothesis and identify relevant genes for biological functions of interest, even for complex traits, which may involve several interacting biological processes and thousands of genes.

3. METHODS

This chapter starts by giving an overview of the DLS method (Puelma et al., 2012). Then, the following sections explain in detail the DLS method and the GENIUS web server.

The main goal of the DLS algorithm is to predict functional associations between genes for a given biological function of interest. DLS is a supervised machine learning method. Supervised methods have to be trained with a set of samples labeled in categories, which are used to learn patterns or mathematical functions that can distinguish, and then classify, new samples among the provided categories. DLS uses this strategy to learn gene expression patterns that are present specifically among genes of a biological function of interest. We call these patterns “expression signatures”, because they identify genes involved in this biological function and distinguish them from those that are not. For this, DLS uses a training set of genes labeled in one of three categories: positive, negative or unlabeled. Positive and negative genes correspond to genes that are involved and not involved in the biological function of interest, respectively (see Section 3.1 below for details), while unlabeled genes are those that cannot be categorized up front as positive or negative with enough confidence. These unlabeled genes can be incorporated in the inferred network if they are classified as positive genes by the prediction algorithm (Figure 3.1).

Briefly, the core of the DLS training algorithm consists of searching an expression signature for each probe matching a positive gene (Section 3.2). For each of these positive probes, DLS searches for a subset of features (experimental conditions) containing a highly discriminative expression pattern (Figure 3.1). The discriminative level of each expression pattern is evaluated by the “expression signature score”, which favors expression patterns having high coexpression with positive probes and penalizes the ones having high coexpression with negative ones. Only the expression patterns that are discriminative enough are considered expression signatures (Figure 3.1). These signatures are then used by DLS to infer a network containing coexpression associations between positive genes and between positive and unlabeled genes (Figure 3.1). Thus, the unlabeled genes that appear in the inferred network correspond to genes that have been predicted by DLS as related to the biological function of interest.

The classification method of DLS uses each expression signature as an independent Bayesian classifier (Section 3.3). Each of this classifiers acts like an expert in identifying positive probes, evaluating its confidence about other probes being positive. The confidence of a signature probe about another probe being positive is based in how correlated (similar) their expression patterns are and how this correlation compares with the ones that the signature has with other positive and negative probes. Probes having a correlation value closer to the ones obtained by positive probes will have higher confidence than those closer to the ones obtained by negative probes. Then, an association (edge) from a signature gene g_i towards a gene g_j is made if and only if the confidence of g_i about g_j being positive is higher than threshold τ.

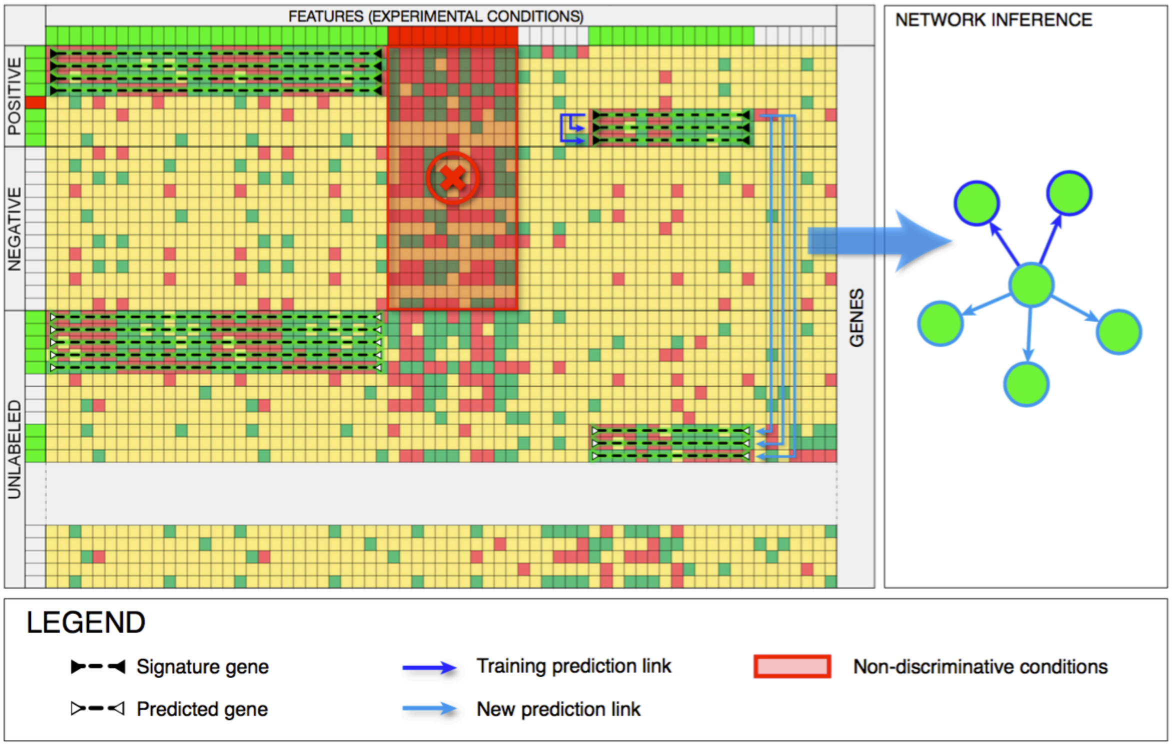


Figure 3.1. Mockup of a labeled training set illustrating the bases of DLS. Each colored square represents the expression level of a gene (row or sample) in an experimental condition (column or feature). DLS searches for expression signatures (ES) for each positive gene. ES are patterns defined by features where positive genes are coexpressed, but positive and negative genes are not (red area). These ES are then used to find genes associated to the positive class by searching genes coexpressed with an ES. Then, each association is represented by an edge in a network.

The rest of this chapter is organized as follows. Section 3.1 presents the methodology used to construct a labeled expression dataset. Sections 3.2 and 3.3 present the main details of the training and classification processes, respectively. Section 3.4 presents an incremental process used in DLS to predict potential false negatives, one of the main problems in using supervised learning to predict gene function. Finally, Section 3.5 describes the GENIUS web server and the features it provides over DLS. In particular, it presents the procedure used to map microarray probes to genes, the server architecture, the datasets and organism included, and the scores and graph theory indicators used for gene prioritization.

3.1. Construction of a labeled expression dataset

A key aspect to perform effective gene functional predictions using massive microarray gene expression data is to apply suitable pre-processing steps to extract informative features and to handle the noisy nature of raw expression data. Also, a key aspect for a supervised method like DLS is to obtain a labeled training set of samples, which we obtain by using the functional knowledge available in Gene Ontology (GO). On one side, the result of pre-processing the gene expression data is a dataset stored in the form of two matrices: one that stores quantitative differential expression values and another that stores the qualitative statistical significance of them. On the other side, the result of the labeling process corresponds to a training set containing a subset of genes labeled either as positive or negative for a particular biological function of interest. The genes known to be involved in this biological function are labeled as positive, whereas the genes not functionally related to the biological function are labeled as negative. The following two sub-sections describe in detail these preprocessing steps.

3.1.1. Pre-processing of gene expression data

Unless microarrays are appropriately normalized, comparing microarray samples from different experiments can lead to misleading results (Irizarry et al., 2003). To overcome this problem, we apply two widely used algorithms: RMA (Irizarry et al., 2003) and RankProducts (Breitling, Armengaud, Amtmann, & Herzyk, 2004). On one hand, RMA uses quantile normalization to make the distribution of expressions comparable among different microarray slides (Bolstad, Irizarry, Astrand, Speed, & Astrand, 2003; Irizarry et al., 2003). On the other hand, RankProducts provides a statistical methodology to find the significance level between expression changes of genes over two experimental conditions containing replicates (Breitling et al., 2004).

In order to describe the pre-processing method, we consider a generic and typical case of public microarray databases, in which we have multiple microarray series or experiments, each coming from different sources, but all from a unique microarray platform. Each experiment corresponds to a series of microarray samples, measured in various, but related, experimental conditions. In addition, each experimental condition has two or more replicates to provide statistical significance to the results. Considering this scenario, we apply the following procedure to the samples of each experiment or series. Given a particular series, the procedure first uses the RMA algorithm over all of its samples to normalize their values. Then, it groups samples that are replicates of the same experimental condition and applies Rank Products to each possible pair of conditions within the series (Figure 3.2). For an experiment with M_C different experimental conditions and coming from a platform with N genes, this procedure generates two matrices with $M_C(M_C\text{-}1)/2$ features in their columns and N genes in their rows. The first matrix (X_{LR}) contains features that provide differential expression values between two experimental conditions. These values correspond to the logarithm of the fold change (log-ratio) of the genes expression levels between two particular experimental conditions. In addition, the second matrix (X_{FDR}) provides the statistical significances of the differential expression values provided in X_{LR}, which are given in the form of false discovery rates (FDRs). In few words, a small FDR value indicates that the corresponding change has a highly consistent rank among the replicates of the compared experiments and thus a low probability of being a false-positive detection (Breitling et al., 2004). The X_{FDR} matrix is used by DLS to guide the search of discriminative expression patterns in X_{LR}, by favoring the features with significant expression changes. Although this procedure might generate some biologically meaningless comparisons, they should not affect the performance of DLS because of its automatic selection of discriminative features, which should filter out non-informative features. Moreover, some unexpected comparisons could still provide new biological insights about the predictions and the biological process of interest.

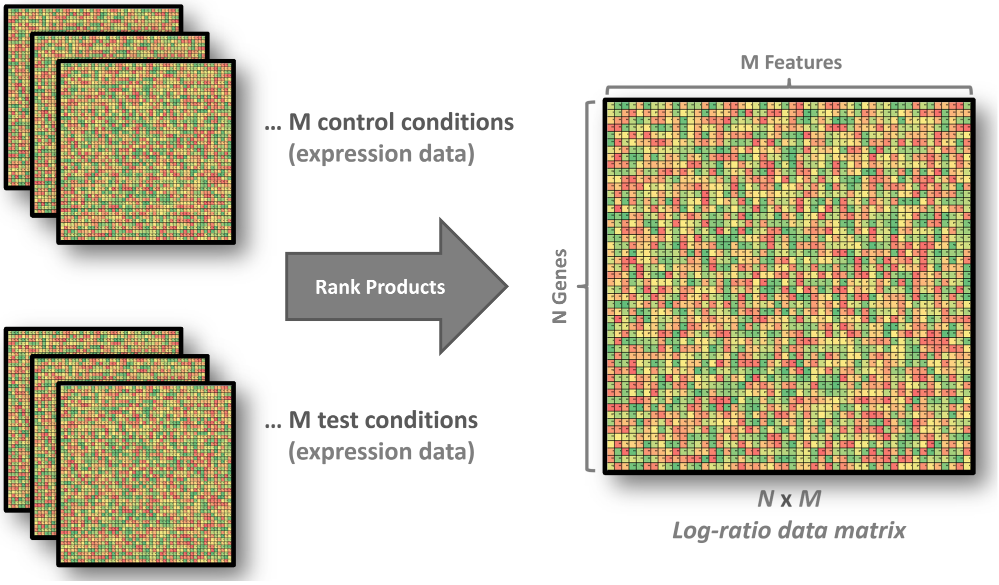


Figure 3.2. RankProducts normalization is used over control-test pairs of conditions to generate a log-ratio differential expression matrix.

In order to minimize the redundancy that this procedure might generate, we only add a feature to the final dataset if it does not have a 'high' correlation with any of the features already added of the same experiment. In the datasets used in this work, we define as 'high' correlations greater that 0.9.

The procedure described above can be independently applied to different platforms of the same organism, generating a row vector with an expression profile for each probe of each platform, which are then matched to genes using the metadata provided, as described in detail in Section 3.5.3. This allows for probes of two different platforms to be matched and grouped to a common gene.

This procedure marks a departure from the pre-processing procedures used in many state-of-the-art methods (for example Alon et al., 1999; Barutcuoglu et al., 2006; Ben-Dor, Chor, Karp, & Yakhini, 2003; Brown et al., 2000; Y. Cheng & Church, 2000; Eisen et al., 1998; Horan et al., 2008; S. K. Kim et al., 2001; Kuramochi & Karypis, 2001; Lanckriet, Deng, Cristianini, Jordan, & Noble, 2004; I. Lee et al., 2010; Mateos et al., 2002; Stuart et al., 2003; Vandepoele et al., 2009; Yang, 2004; Zhang et al., 2004; Zhao et al., 2008), which base their inferences on absolute gene expression data, as the one provided by RMA. The differential expression data provided by RankProducts allows us to filter out most of the artifactual expression patterns, leaving only patterns that show statistically significant changes. In addition, it diminishes the biases introduced in each series by comparing the amount of variation in gene expression, which allows us to combine features coming from different series. Finally, it allows to automatically pair experimental conditions to measure differential expression, which is a tedious task that is commonly done manually by an expert, but impractical for massive datasets containing hundreds or even thousands of experimental conditions (See Section 3.5.2 for a description of the datasets used for the GENIUS web server).

3.1.2. Training set construction using Gene Ontology

A critical aspect for a supervised prediction approach is the construction of a labeled training set. As a consequence, DLS needs a labeled training set of genes in order to search for discriminative expression patterns for a specific biological function of interest (BF). Each training gene must be labeled as positive or negative, depending on whether the gene participates or does not participate in BF, respectively.

Manually defining the gene set for training can be a difficult and tedious process. In particular, negative training genes are difficult to define, due to the almost total absence of negative annotations, our incomplete knowledge and the ability of genes to have multiple roles. However, DLS (and GENIUS) makes this process transparent for the user. In order to make a prediction, the user only has to define a query list with positive genes for the biological function of interest. In particular, the GENIUS web server provides two complementary options for this process. The first option is to submit a custom list of genes or probes identifiers. GENIUS will then search for all genes in its database that matches any of the provided identifiers (See Section

3.5.3 for details about the mapping of probes to genes). The second option to define the query list is to select one or more Gene Ontology (GO) biological processes related to the biological function of interest. GENIUS will then add to the query list all the genes annotated in any of the selected GO terms.

Once the positive set is defined, DLS generates a plausible set of negative genes by using an adapted version of the Rocchio algorithm, which was originally developed for text classification (Rocchio, 1971), but adapted for negative protein function prediction (Youngs, Penfold-Brown, Bonneau, & Shasha, 2014). This method uses GO annotations to derive a score that measures the semantic distance of each gene to the positive set. Thus, the higher the score of a gene, the less related to the positive genes it is according to its annotations, and thus, to the biological function of interest. An inconvenience of this method is that it needs a score threshold to define which genes are considered negative. Based in our tests, we obtain a suitable number of negative genes for DLS by using the top 50% of genes that have a score greater than zero. Note that using this criterion, DLS is able to slightly improve its prediction performance as compared to the method incorporated in the former published DLS algorithm (Puelma et al., 2012).

3.2. Training: searching expression signatures

The core of the training process of DLS is to identify suitable expression signatures for the biological function of interest BF. Each expression signature is defined by a discriminative local subspace of the expression data matrix X_{LR} described in Section 3.1.1. The core of this scheme is based on four concepts about gene expression:

i. Coexpression: genes exhibiting coexpression patterns are likely to be coregulated, and hence, they are likely to participate in a common biological function. Consequently, DLS uses the positive set of genes C^{1}_{BF} to search for common coexpression patterns for genes involved in BF (Figure 3.3).

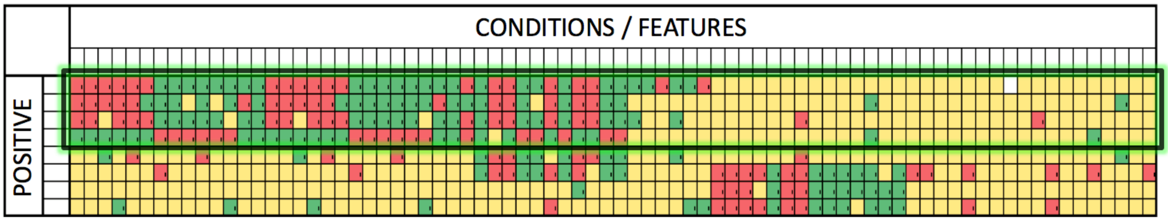


Figure 3.3. Coexpressed genes usually participate in a common biological process.

ii. Subspaces: genes participating in the same biological function are usually not coregulated under all the cellular conditions. Consequently, DLS searches for patterns under subsets of conditions that maximize coexpression among positive genes (Figure 3.4).

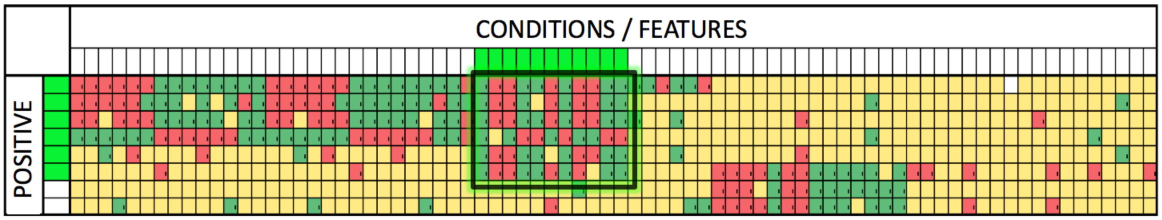


Figure 3.4. Genes are usually coexpressed under subsets of conditions (features).

iii. Locality: genes participating in the same biological function may be regulated by different transcription factors and hence, they might be coexpressed under different experimental conditions. Consequently, DLS independently searches for a suitable subset of conditions for each positive gene in C^{1}_{BF} (Figure 3.5).

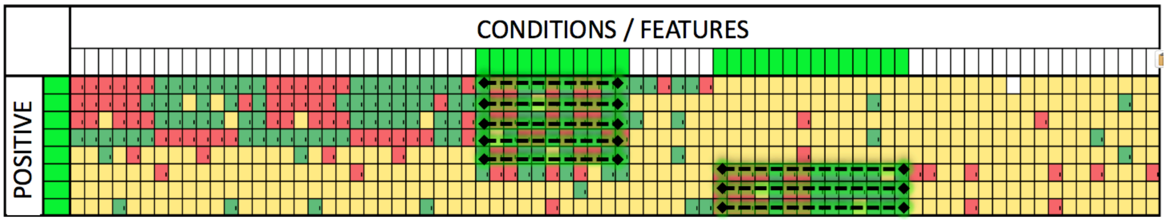


Figure 3.5. Genes can be coexpressed under different subsets of conditions.

iv. Discrimination: genes participating in different biological functions may still coexpress under some experimental conditions. Consequently, DLS uses the negative genes C^{0}_{BF} to filter out non-discriminative subsets of conditions where positive and negative genes show coexpression patterns (Figure 3.6).

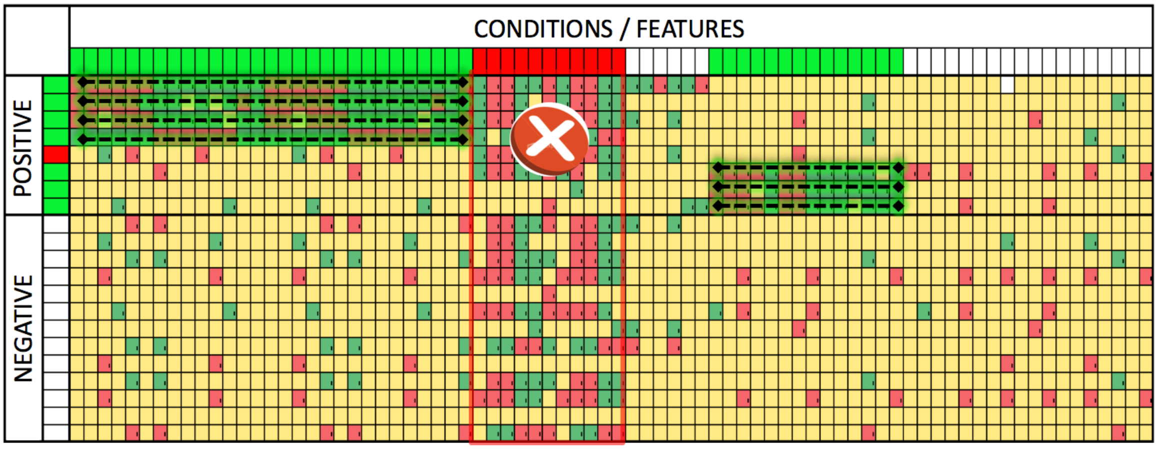
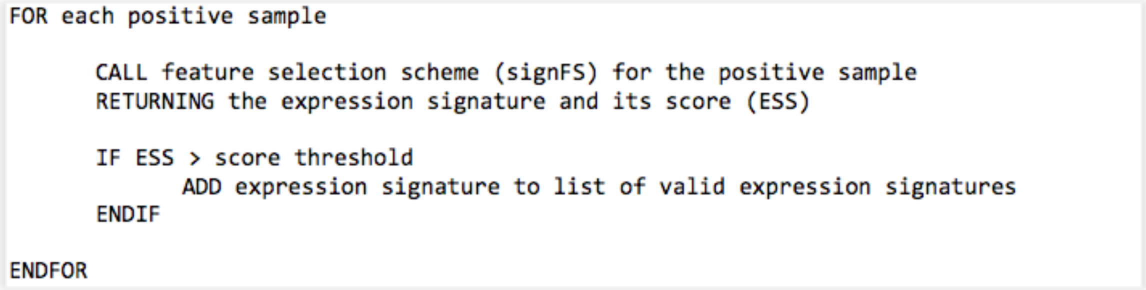


Figure 3.6. Not all conditions are discriminative to the target biological process.

In agreement with the previous concepts, the core of the training process consists of a feature selection algorithm that looks for a suitable expression signature for each gene $g_i \in C^{1}_{BF}$ (Figure 3.7). We achieve this by selecting a subset of features where g_i shows a 'strong' coexpression with genes in C^{1}_{BF} and a 'weak' coexpression with genes in C^{0}_{BF}. This feature selection algorithm explores the space of possible subsets of features using the Expression Signature Score (*ESS*) presented in equation (3.1) below. This score evaluates the discriminative power of each potential subset (pattern). Once the feature selection scheme is finished, each positive gene $g_i \in C^{1}_{BF}$ has an associated subset of features corresponding to the most discriminative expression pattern found by DLS. However, only expression patterns having an $ESS > 0$ are selected as valid expression signatures and used in the classification process. We describe next the details of the *ESS* score and then the main steps behind the operation of the feature selection scheme, which we refer to as *signFS* (Figure 3.8).



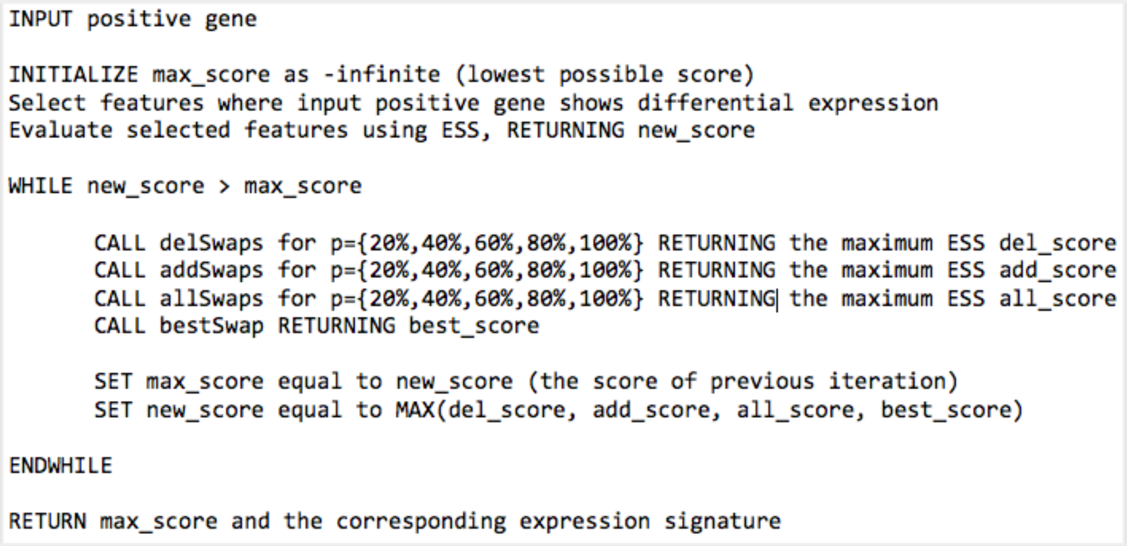
```
FOR each positive sample

      CALL feature selection scheme (signFS) for the positive sample
      RETURNING the expression signature and its score (ESS)

      IF ESS > score threshold
            ADD expression signature to list of valid expression signatures
      ENDIF

ENDFOR
```

Figure 3.7. Pseudocode of the training algorithm.



```
INPUT positive gene

INITIALIZE max_score as -infinite (lowest possible score)
Select features where input positive gene shows differential expression
Evaluate selected features using ESS, RETURNING new_score

WHILE new_score > max_score

      CALL delSwaps for p={20%,40%,60%,80%,100%} RETURNING the maximum ESS del_score
      CALL addSwaps for p={20%,40%,60%,80%,100%} RETURNING the maximum ESS add_score
      CALL allSwaps for p={20%,40%,60%,80%,100%} RETURNING the maximum ESS all_score
      CALL bestSwap RETURNING best_score

      SET max_score equal to new_score (the score of previous iteration)
      SET new_score equal to MAX(del_score, add_score, all_score, best_score)

ENDWHILE

RETURN max_score and the corresponding expression signature
```

Figure 3.8. Pseudocode of the feature selection scheme (*signFS*).

3.2.1. Expression signature score

Let f_{sel} be a selected subset of the total set of available features. Furthermore, let $g_i[f_{sel}]$ be the expression pattern of gene g_i considering only the features in f_{sel} (i.e. $g_i[f_{sel}] = X_{LR}(i, f_{sel})$)). The *ESS* of gene g_i for a subset of features f_{sel} is defined as

$$ESS(g_i[f_{sel}]) = w_1 \cdot Score_1(g_i[f_{sel}]) - w_0 \cdot Score_0(g_i[f_{sel}]), \qquad (3.1)$$

where *Score1*$(\cdot)$ and *Score0*$(\cdot)$ are functions that quantify the coexpression level of gene g_i with respect to the set of genes in C^{1}_{BF} and C^{0}_{BF}, respectively, considering only features in f_{sel}. More precisely,

$$Score_*(g_i[f_{sel}]) = \sum_{j \in C^*_{BP}} sgd\left(coexp\left(g_i[f_{sel}], g_j[f_{sel}]\right)\right), \tag{3.2}$$

where *coexp*$(\cdot,\cdot)$ measures the coexpression between two patterns and *sgd*$(\cdot)$ corresponds to a sigmoidal function used to establish a continuous threshold to separate ‘strong’ from ‘weak’ coexpressions (Figure 3.9).

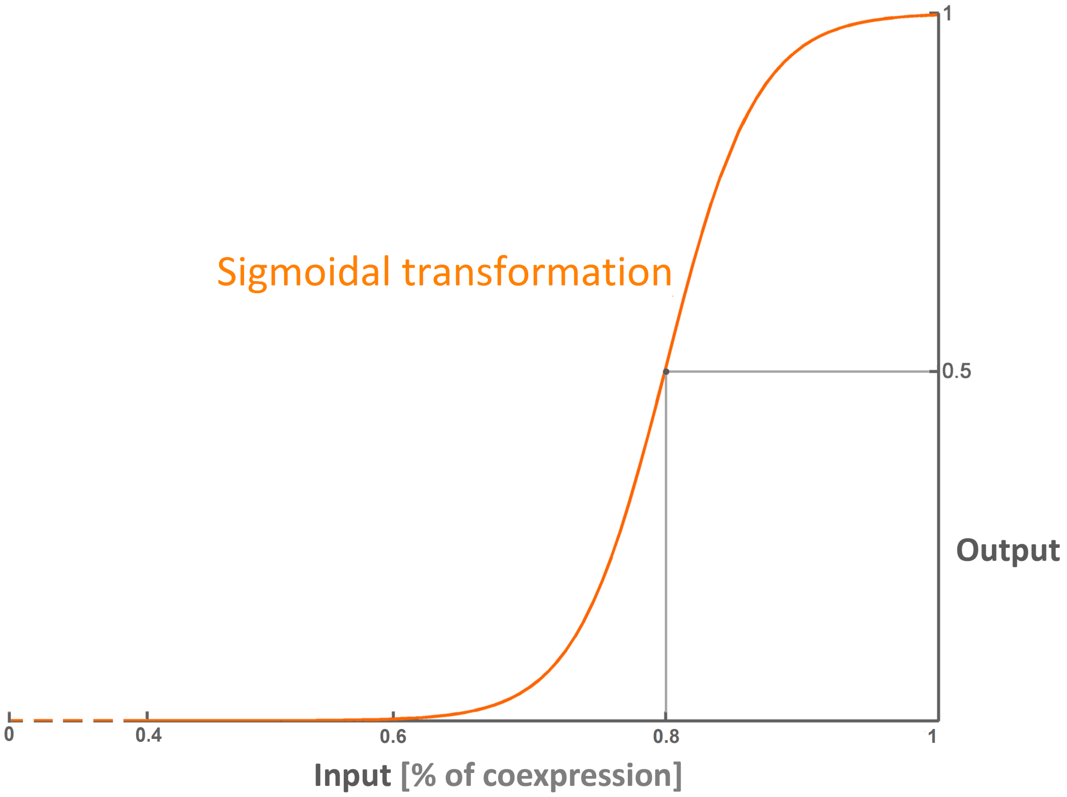


Figure 3.9. Sigmoidal function used to increase the difference between “weak” coexpressions (<0.6) and “strong” coexpressions (>0.8).

The shape of this sigmoidal function (Figure 3.9) was tuned for best prediction performance using tests described in Section 4.1 and taking into account our biological and mathematical knowledge as well. As a result, the function returns values between 0 and 1, being close to 0 for coexpression values below 0.6 (weak) and above 0.5 for coexpression values above 0.8 (strong) (Figure 3.9).

To measure coexpression between the expression patterns of two genes g_i and g_j considering features in f_{sel}, we use the absolute value of the cosine correlation, which can be expressed as the dot product of two vectors, normalized by their respective magnitudes:

$$coexp\left(g_i[f_{sel}], g_j[f_{sel}]\right) = abs(cos\_corr(g_i[f_{sel}], g_j[f_{sel}])) \qquad (3.3)$$

$$coexp\left(g_i[f_{sel}], g_j[f_{sel}]\right) = abs\left(\frac{g_i[f_{sel}] \cdot g_j[f_{sel}]}{\|g_i[f_{sel}]\|\|g_j[f_{sel}]\|}\right)$$

The cosine correlation (*cos_corr*) returns a continuous value between 1 and -1, taking a value of 1 if the two patterns are correlated, -1 if they are negatively correlated and 0 if they change independently. We use the absolute value *abs*(·) to capture positive and negative correlations indistinctively among genes, which improves the prediction performance in our test. Despite its simplicity, we consider this measure more suited than the traditional Pearson correlation coefficient (PCC) to measure coexpression in log-ratio expression data, in which each feature is a comparison in itself between two conditions. This can be more clearly seen by the following example: consider the log-ratio expression patterns of genes $g_1 = [1,1,0,0]$ and $g_2 = [0,0,-1,-1]$. Analyzing these two patterns, we intuitively do not expect any relation between their corresponding genes because the expression of gene g_1 is not affected at all when gene g_2 changes (i.e. $g_1 == 0 \leftrightarrow g_2 \neq 0$) and vice versa. This is very well expressed by the cosine correlation, which returns a value *cos_corr*(*p1*,*p2*) = 0. Contrarily, the PCC only considers the relative changes within the features of the patterns, which in this example are perfectly synchronized, thus returning a $PCC(p_1,p_2) = 1$, the opposite from what we expect.

In equation (3.1), w_1 and w_0 weight the influence of $Score_1(\cdot)$ and $Score_0(\cdot)$, respectively, and are used to avoid overfitting the training samples; w_1 is defined by a function that penalizes expression signatures with a small number of features, whereas w_0 is a predefined parameter that allows us to adjust the level of discrimination of the expression signatures. Specifically:

$$w_1(|f_{sel}|) = \begin{cases} 1 - (w_1\_th - |f_{sel}|) \cdot \Delta w_1 & if \quad |f_{sel}| < w_1\_th \\ 1 & otherwise \end{cases}, \qquad (3.4)$$

where $|f_{sel}|$ is the number of features in f_{sel}. Equation (3.4) penalizes the signatures with less than w_1_th features by lowering the corresponding weight w_1 by Δw_1 for each eliminated feature below w_1_th. The value of w_1_th is adapted for each trained signature, depending on the number of features where the corresponding gene g_i shows significant expression changes $|f_{sign}|$):

$$w_1\_th = round(|f_{sign}| \cdot w_1\_a), \qquad (3.5)$$

where w_1_a is a parameter corresponding to the fraction of significant features that will be used as threshold. Back into equation (3.4), Δw_1 is derived as a linear function such that $w_1(|f_{sel}|) == 1$ if $|f_{sel}| == w_1\_th$ (no penalization) and $w_1(|f_{sel}|) == w_1\_b$ if $|f_{sel}| ==$ *min_feats* (maximum penalization):

$$\Delta w_1 = (1 - w_1\_b)/(w_1\_th - min\_feats), \qquad (3.6)$$

where w_1_b and *min_feats* are parameters corresponding to the minimum w_1 value (maximum penalization) reached when an expression pattern has a minimum number of features *min_feats*.

3.2.2. Feature selection scheme

The feature selection algorithm, *signFS*, uses the *ESS* score in equation (3.1) to find a suitable expression signature for each positive gene $g_i \in C^1_{BF}$ (Figure 3.8). An exhaustive search, however, is not possible because it requires the evaluation of 2^M - 1 possible subsets of features for each positive gene. Consequently, we use an iterative and fast exploration scheme, which uses suitable heuristics to efficiently search for discriminative expression signatures.

Given a gene expression pattern $g_i = X_{LR}(i,:)$, *signFS* obtains an initial set of features, $f_{sel}(0)$, by selecting the features where gene *i* significantly changes its level of expression. We define as significant, a change with a FDR value < 0.1 in $X_{FDR}(i,:)$. Afterwards, *signFS* performs an iterative process (explained below) that, at each iteration *t*, adds and/or removes a suitable subset of features F_t, from $f_{sel}(t)$. These changes must increase the expression pattern score *ESS*(·) (Equation 3.1). As a consequence, the new subset $f_{sel}(t+1)$ should provide better discriminative properties for gene function prediction. This iterative process continues until consecutive modifications of $f_{sel}(t)$ do not increase the respective score *ESS*(·).

A key aspect of the previous iterative process is how to choose a suitable subset of features F_t to swap (add/remove) from $f_{sel}(t)$, such that the resulting subset $f_{sel}(t+1)$ provides better discriminative properties for gene function prediction. The most straightforward option for this would be to apply a greedy stepwise feature selection (SFS) scheme, which adds/removes only the single feature that produces the highest increase in the expression signature score $ESS(g_i[f_{sel}(t)])$ (Equation 1). Modifying only one feature at each iteration, however, can lead to a slow convergence, particularly at early stages of the process, where the current subset $f_{sel}(t)$ is likely to be far from an optimal situation. Furthermore, such local search is likely to be trapped by a local optimum.

To tackle the problems exposed above, we evaluate swaps of more than one feature at each iteration. Each iteration starts by limiting the features available for swapping to the top $\widehat{M}(t)$ most significative features according to the X_{FDR} matrix (i.e. with lower FDR value). $\widehat{M}(t)$ corresponds to the number of features selected at iteration *t* in $f_{sel}(t)$, plus the 20% most significative features not currently selected in $f_{sel}(t)$. This initial restriction allows DLS to foster the inclusion of features showing the most significative changes in expression.

Unfortunately, for subsets of k simultaneous swaps among the $\widehat{M}(t)$ features, the number of possible different patterns P that can be evaluated is given by $P = \widehat{M}(t)!/k!\,(\widehat{M}(t)-k)!$. This has a computational complexity $O(\widehat{M}(t)^k)$, which is exponential in *k*. Thus, we apply a heuristic method that uses the *ESS*(·) score of the $\widehat{M}(t)$ individual swaps (*k* = 1), to efficiently guide the selection of a *good* subset (*k*

≥ 1) of features F_t to swap. By *good* we mean that swapping the features in F_t increases the score of $f_{sel}(t)$ at least as much as the greediest option (k = 1).

The heuristic mentioned above first explores local changes by evaluating the *ESS*(·) score of the $\widehat{M}(t)$ patterns formed by swapping the $\widehat{M}(t)$ available features individually (*k* = 1). Then, to explore global changes, it evaluates 16 additional patterns, formed by swapping 16 subsets of $k \geq 1$ features. These subsets are selected from the $\widehat{M}(t)^* \leq \widehat{M}(t)$ features that increase the score in the previous evaluation with *k* = 1. Finally, from the $\widehat{M}(t)$ + 16 resulting patterns, the one with the greatest increase in the *ESS* score is selected as the set $f_{sel}(t+1)$. If none of them obtains a score *ESS*(·) higher than the current set $f_{sel}(t)$, then the iterative process ends and *signFS* returns the *ESS*(·) score and the current selected features.

The heuristic explained above allows DLS to explore local and global changes by evaluating only 16 different subsets of swaps per iteration, whose features are selected by four different methods: *delSwaps*, *addSwaps*, *allSwaps*, and *bestSwap*. Let F_{inc} be the subset of features that increases the *ESS*(·) score of $f_{sel}(t)$ with a single swap (*k* = 1).

i. delSwaps removes from $f_{sel}(t)$ the p% of the features present in both, F_{inc} and in $f_{sel}(t)$, whose resulting subsets show the greatest increase in ESS(·) score.
ii. addSwaps adds to $f_{sel}(t)$ the p% of the features in $F_{inc} \notin f_{sel}(t)$, whose addition produces subsets with the greatest increase in ESS(·) score.
iii. allSwaps(p) considers all the features in F_{inc}, and from those, swaps the p% with greatest score.
iv. bestSwap swaps only the feature from F_{inc} with the greatest score.

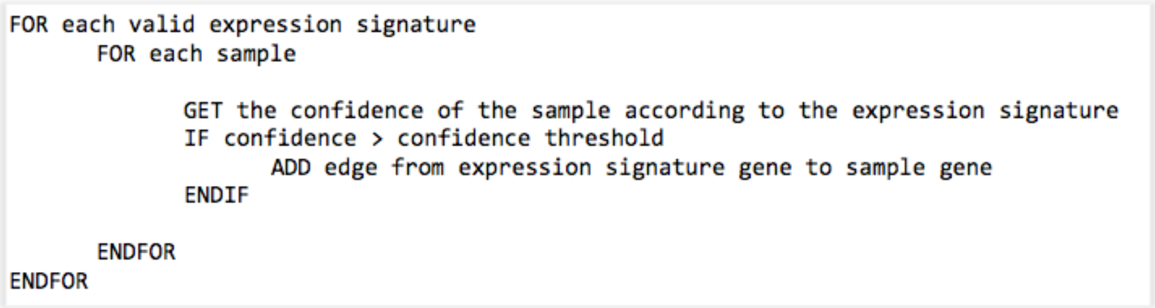
The first three methods are applied using 5 different percentages p = {20, 40, 60, 80, 100}, which allows us to explore local and global changes (Figure 3.8).

Consequently, the swaps defined by the four methods generate 5 + 5 + 5 + 1 = 16 new potential expression patterns. These patterns are then evaluated and the one with greatest *ESS* score is selected as the set $f_{sel}(t+1)$. If none of them obtains a score

ESS$(\cdot)$ higher than the current set $f_{sel}(t)$, then the iterative process ends and *signFS* returns the *ESS*$(\cdot)$ score and the current selected features (Figure 3.8).

3.3. Classification: using expression signatures to predict gene associations

The aim of the classification scheme used by DLS is to use the expression signatures found during training in order to predict functional gene associations for the biological function of interest BF. This section starts describing the details behind the classification algorithm to predict these associations. Then, it describes how DLS uses these predicted associations to form a gene network focused in the genes containing expression signatures and the genes predicted to be associated to them (Figure 3.10).



```
FOR each valid expression signature
      FOR each sample

            GET the confidence of the sample according to the expression signature
            IF confidence > confidence threshold
                  ADD edge from expression signature gene to sample gene
            ENDIF

      ENDFOR
ENDFOR
```

Figure 3.10. Pseudocode of the classification algorithm.

3.3.1. Prediction of gene associations

Expression signatures are trained to be discriminative. Thus, if a gene g_i is highly coexpressed with the expression signature of a gene g_j in C^{l}_{BF}, then g_i is likely to be involved in BF (i.e. there is a functional association between g_i and g_j). A relevant issue with respect to the previous classification scheme is that not all the expression signatures have the same potential to predict functional associations. In effect, this potential depends on several factors such as type of gene, type of biological function, biological complexity of interprocess co-regulations, and level of noise in the data. DLS overcomes these issues by using a Bayesian inference approach that allows it to adaptively decide the minimum coexpression level needed by each signature to predict a gene with a given confidence (Figure 3.11).

Consider a hypothesis h, representing that an unknown gene g_j belongs to the positive class $C^1{}_{BF}$. In addition, consider evidence e, indicating that gene g_j has a coexpression level L, with respect to the expression signature of gene g_i, $ES(g_i)$. We can estimate the posterior probability $P(h|e)$ by using Bayes rule:

$$P(h|e) = \frac{P(e|h)\times P(h)}{P(e)} \tag{3.7}$$

where

$$P(e) = P(e|h)P(h) + P(e|\neg h)P(\neg h) \tag{3.8}$$

Thus, we need to calculate basically two types of probabilities: the prior probabilities, $P(h)$ and $P(\neg h)$, and the positive and negative likelihoods, $P(e|h)$ and $P(e|\neg h)$, respectively. Prior probabilities can be estimated directly from training data by calculating the proportion of positive versus negative genes in the training set. However, the estimation of the likelihood terms is not so straightforward as we need to estimate the probability density function (pdf) of the coexpressions with respect to $ES(g_i)$. To overcome this, we estimate the likelihoods using a kernel-based density function estimation (Parzen, 1962), which can be thought as a continuous and smoothed version of a histogram estimation. Given a particular coexpression x_i and a bandwidth σ, we use a Gaussian kernel function $K(x) = N(x_i, \sigma)$, which measures the influence of sample x_i in a location x of the input space. The bandwidth σ is a parameter that controls the smoothness of the pdf. Then, the total density in any location x can be estimated by summing the influences (kernel values) at location x of all the available samples x_i. In this particular case, the input space corresponds to the space of coexpression levels that training genes have with signature $ES(g_i)$ (Figure 3.11A).

The choice of bandwidth σ usually has a great impact in the resulting estimation, in a similar way to the effect of bin size in a histogram pdf estimation. On one hand, a small σ could produce an under-smoothed pdf estimation containing spurious data artifacts. On the other hand, a large σ could produce an over-smoothed density that may obscure the underlying density shape. To tackle this problem, we use a kernel bandwidth optimization method (Shimazaki & Shinomoto, 2010) that automatically selects the optimal bandwidth σ for a given set of data samples and a kernel function. Our experiments have shown that the best results are obtained by restricting the selected bandwidth to the range [0.08, 0.1]. Thus, we use the bandwidth $\sigma*$ given by

$$\sigma^* = \min\left(\max(sskernel(\cdot), 0.08), 0.1\right) \tag{3.9}$$

where *sskernel*$(\cdot)$ function returns the bandwidth given by the optimization method, and *max*$(\cdot,\cdot)$ and *min*$(\cdot,\cdot)$ functions return the maximum and minimum values among the inputs, respectively.

By applying the Bayes procedure described above, a gene g_j is predicted as positive by an expression signature $ES(g_i)$ if the coexpression L between them results in a confidence $P(h|e)$ greater than a desired threshold (Figure 3.11D). Notice that although a fixed confidence threshold is needed, it is translated into a different coexpression threshold for each signature, depending on the evidence shown in the training set about its discriminative potential (Figure 3.11C-D).

We encourage the reader to see the example in Figure 3.11 to get a better understanding of the classification process. The figure illustrates of the method used to determine the correlation threshold and the prediction confidences given by a signature. The signature in the example corresponds to the *A. thaliana* gene HVA22D (AT4G24960) of the response to abscisic acid stimulus (GO:0009737) biological process. In Figure 3.11A, the positive and negative likelihoods ($P(e|[\neg]h)$) are based on the density of the coexpressions of the positive and negative training genes with the signature. In this case, the likelihood for coexpressions greater than 18% are higher for positive than for negative genes, as expected for a discriminative pattern. In Figure 3.11B, the negative training set is much larger than the positive set (1917 versus 208 genes), so there is a strong prior in favor of the negative class ($P(\neg h) = 0.9$ and $P(h) = 0.1$). Thus, after weighting by the respective priors, the positive class becomes less likely. In Figure 3.11C, by zooming in, it can be seen that now the positive class is more likely than the negative class only for coexpressions greater than 64%. In Figure 3.11D the weighted likelihoods are then used to determine the posterior probability function ($P(h|e)$), which gives the confidence to the predictions made by this signature. As a discriminative signature, it will provide higher prediction confidences to genes showing higher coexpressions with it. In particular, the minimum reasonable coexpression threshold for this signature is 64%, as genes with lower coexpressions would be more likely to be negative than positive, based on the training evidence.

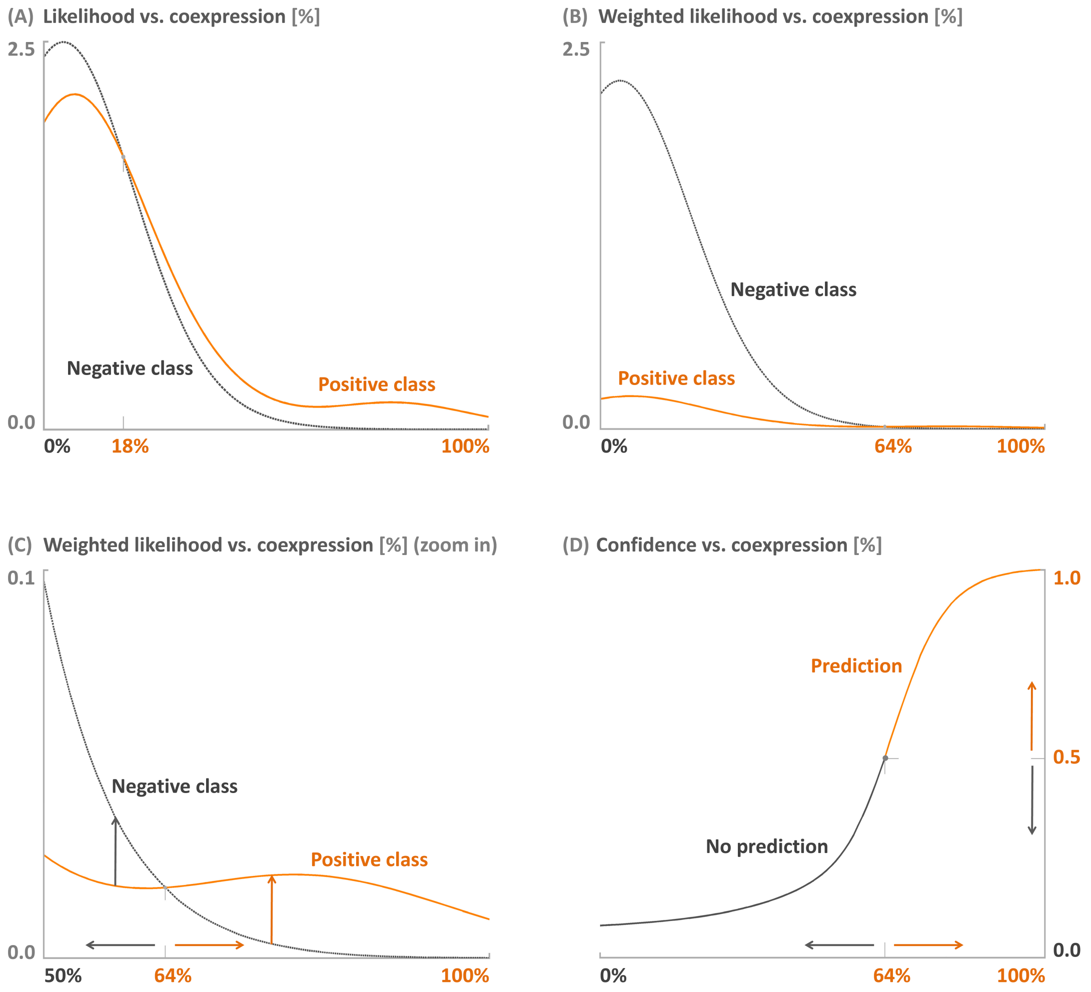


Figure 3.11. Example of the Bayesian inference approach used in DLS to determine the correlation threshold and the prediction confidences given by a signature. (A) Positive and negative likelihoods *P(e/[¬]h)* are based in the density of coexpressions of positive and negative training genes with the signature. In this case, the likelihood for coexpressions greater than 18% are higher for positive than for negative genes, as expected for a discriminative pattern. (B) The negative set is much larger than the positive set, so there is a strong prior in favor of the negative class (*P(¬h) >> P(h)*). Thus, after weighting by the respective priors, the positive class becomes less likely. (C) A zoom in shows that the positive class is still more likely than the negative one for coexpressions greater than 64%. (D) The weighted likelihoods are then used to determine the posterior probability function (*P(h/e)*) of this signature, which gives the confidence to the predictions made by this signature. As a discriminative signature, it gives higher prediction confidences to higher coexpressed genes. In particular, the minimum coexpression threshold for this signature should be 64%, as genes with lower coexpressions are more likely to be negative than positive.

In GENIUS, we set the confidence threshold τ to be 0.66, so that the confidence of a predicted gene g_j to be positive is at least twice of the confidence about it being negative (0.66 vs 0.33). However, DLS can lower or increase this threshold dynamically in order to obtain a reasonable number of genes in the final network. Let *#positives* be the number of positive genes in the training set and *#predictions* the number of genes with a confidence higher than τ. Then, threshold τ is lowered if $(\#predictions < (\#positives)/2)$ & $(\tau > 0.5)$ or increased if $(\#predictions > 2 * \#positives)$ & $(\tau < 1.0)$. The restriction of $\tau > 0.5$ allows DLS to infer networks enriched and focused in genes that have a higher probability of being related to BF than not being involved in it.

3.3.2. Gene network derivation

One of the main features behind DLS is its ability to represent its predictions as a discriminative coexpression network (DCN), which provides additional insights about the predictions and the biological function of interest. Formally, a DCN for a biological function BF is defined by a graph $G_{BF} = < V, E >$, where vertices in set V represent genes, and edges in set E represent predicted associations from expression signatures genes to other genes. More precisely, there is an edge from gene $g_i \in C^{l}{}_{BF}$ to gene g_j, if there is an expression signature $ES(g_i)$ predicting that g_j is related to BF with a confidence greater than a pre-defined threshold (Figure 3.11D). In order to construct a DCN that involves all the genes related to BF, DLS applies the classification algorithm to all the N genes available in matrix X_{LR}, including the ones in C_{BF} used for training. This not only allows DLS to display a network description of the relations between training genes and new predicted genes, but also to expose relevant relations among the positive genes, even if they are known to be involved in BF. A network description allows application of tools and concepts (Strogatz, 2001) developed in fields such as graph theory, physics and sociology that have dealt with network problems before (Alon, 2003). For example, GENIUS calculates the node degree and betweenness centrality of genes in the DCN, which provide relevant insights to discover central and highly coordinated genes in the biological function of interest. Details about these indicators are found in Section 3.5.4. In addition, Section 4.4.2 shows a case study that demonstrate the usefulness of these indicators.

There are various characteristics that distinguish DCNs from traditional coexpression networks (CNs). On one side, there is a semantic difference between the links of both networks. In CNs, a link between two genes is undirected and represents that both genes are coexpressed above some given and fixed correlation threshold. In contrast, in DCNs a link is directed and represents that a signature gene g_i predicts gene g_j with a confidence above some given and fixed confidence threshold. Defining the coexpression threshold is one of the main difficulties in constructing CNs. Different genes can show coexpression patterns in different subsets of conditions, thus varying the optimal global correlation threshold. In contrast, the Bayesian inference approach used by DLS allows it to select the coexpression threshold adaptively for each expression signature. On the other side, in CNs, coexpression is measured over all available experimental conditions. In contrast, in DCNs, it is measured over subsets of discriminative conditions, which are selected differently for each expression signatures, so that they are differentially expressed and have a high coexpression level with other genes involved in BF. This allows DCNs to show connections that are hidden among some specific conditions and to filter out noisy and irrelevant conditions.

3.4. False-Negatives Discovery: overcoming the false negatives problem

One of the most relevant issues of using supervised learning methods to predict gene function is the false negatives (FNs) problem. In Section 3.1.2 we show a method to obtain an informative negative training set C^0_{BF}, containing genes that according to GO annotations have a low chance of being related to the positive class C^1_{BF}. Unfortunately, due to the inherent complexity of gene behavior and the incompleteness of annotations, it is not possible to know with certainty which genes are not involved in a biological function. As a consequence, we expect for a proportion of the negative training genes to be mislabeled, which can damage the performance of the predictions. As FNs are actually positive genes, they are likely to be coexpressed with other positive genes, masking the discriminative expression patterns.

To tackle the previous problem, we add to our training algorithm the option of a bootstrap step, which is able to automatically identify, and temporarily discard from the set C^{0}_{BF}, genes that are potential FNs and are obscuring the finding of discriminative patterns. More specifically, this strategy is applied at the start of each step *t* of the feature selection algorithm *signFS*, performed in the training of each positive gene g_i. The strategy discards a negative gene g_j from iteration *t* if its coexpression with gene g_i, using the selected features in $f_{sel}(t)$, satisfies two conditions:

i. it is among the top pFN% most highly coexpressed negative genes and
ii. it has a value of at least min_FN_coexp.

Notice that these potential FN genes are not discarded permanently from the negative training set, but they are only not considered in the evaluations of the patterns generated during the step *t*. At the end of the training process, the method outputs the potential FNs detected by each expression signature.

The bootstrap option explained above allows us to avoid overfitting problems due to the presence of FNs in the training set, however, it may affect the discriminative level of the signatures by ignoring some true negatives during the training process. Thus, we developed an iterative method, False-Negatives Discovery (FND), which takes advantage of this option in order to predict FN genes in a more precise and informative manner. Initially, the list G_{FN} of potential FNs contains all negative genes in C^{0}_{BF} ($G_{FN} = C^{0}_{BF}$). Then, each iteration of the method applies three consecutive steps, used to incrementally bound and refine the list G_{FN}. In the first step, a complete model is trained using the bootstrap option explained in the previous paragraph. G_{FN} is then bounded to the potential FN genes detected by at least one trained signature in these model. In the second step, the trained signatures are used to classify the genes in G_{FN}, filtering out the ones not predicted as positive. Finally, in the third step, the training algorithm is used to search for a suitable expression signature for each gene in G_{FN}. This algorithm is used without the bootstrap option. Then, a gene g_j in G_{FN} is predicted as a FN if the method is able to find an expression signature $ES(g_j)$ that satisfies two conditions:

i. $\mathrm{ESS}(\mathrm{g_j}[\mathrm{f_{sel}}]) > 0$ (Equation 3.1) and

ii. $\mathrm{Score_1}(\mathrm{g_j}[\mathrm{f_{sel}}])$ (Equation 3.2) is greater than the average $\mathrm{Score_1}(\cdot)$ obtained among the valid expression signatures of the positive class C_1^{BF}.

The first condition imposes the predicted FNs to be discriminatively connected to other positive genes, whereas the second condition imposes them to be at least as connected as an average positive gene.

The three steps described earlier are executed iteratively by the FND method, automatically moving the predicted FNs to the positive set of the next iteration. The method stops if no new FNs are predicted or if a maximum number of iterations are reached. After performing the FND method, the training set can be refined, either by eliminating the predicted FNs from the negative set or by moving them to the positive set. This refined set is then used to train a DLS model and obtain the final predictions.

3.5. GENIUS: a user-friendly web interface for the DLS method

GENIUS (GEne Networks Inference Using Signatures) is a web server with a user-friendly interface to DLS that allows the scientific community to fully exploit its capabilities. GENIUS incorporates Gene Ontology annotations and thousands of microarrays experiments from Gene Expression Omnibus (GEO) for eight model organisms. In addition, GENIUS adds several tools to visualize and analyze gene networks, including integration with Cytoscape (Smoot et al., 2011), an advanced network analysis software platform. The following subsections describe GENIUS web server and its features in detail.

3.5.1. Server architecture

GENIUS web server is composed by three main layers: user interface (UI), web integration, and data processing (Figure 3.12). The UI layer presents the data in the client’s browser using Adobe Flash technology and was developed using the Adobe

Flex framework. The web integration layer composed by an Apache Tomcat server, a MySQL database, and the web application, which was developed using Java 1.6. The web application mediates the communication of the client with the database and the data processing layer. It also does some minor processing, like search queries, generation of output files, and system reports. Finally, the data processing layer is composed by a series of Java services that call the MATLAB processes that run the DLS algorithm and make the network inferences. These services communicate asynchronously with the web integration layer using Java Remote Method Invocation (RMI). With this approach, users do not require waiting online for the results.

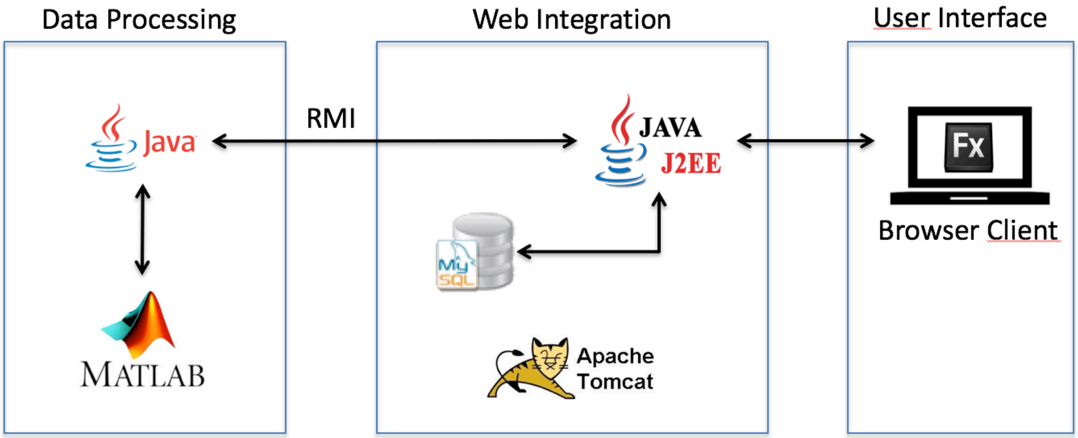
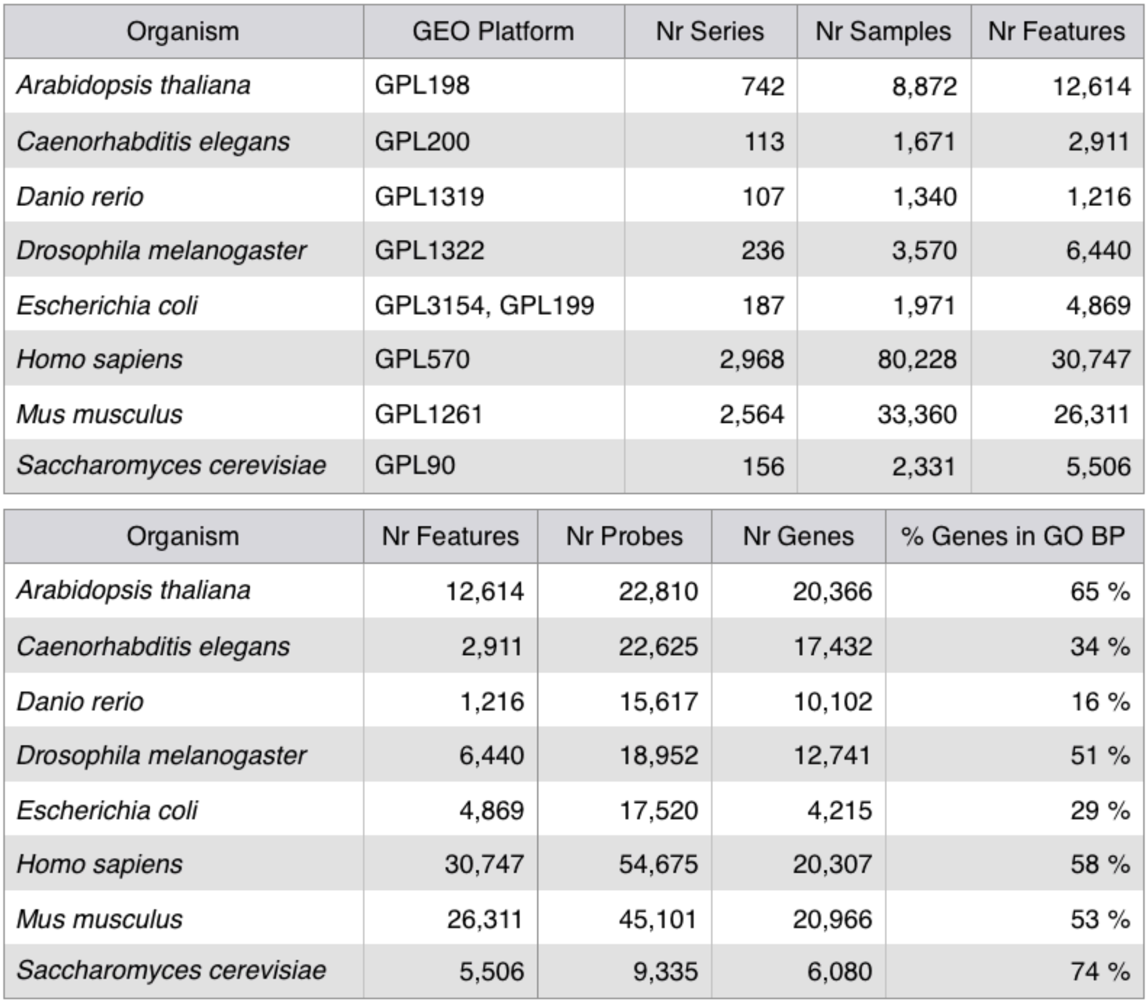


Figure 3.12. GENIUS server architecture has three main layer: data processing, web integration, and user interface. Data processing is done mainly by the DLS algorithm implemented in MATLAB. Also, some minor tasks are performed using Java. The web integration layer mediates the communication of the client with the database and the data processing layer. Finally, the user interface layer provides the graphic interface and web application that users interact with.

3.5.2. Datasets and supported organisms

GENIUS incorporates microarray gene expression data from Gene Expression Omnibus (GEO) (Edgar, Domrachev, & Lash, 2002) and gene functional annotations from Gene Ontology (GO) (Ashburner et al., 2000) for eight model organisms: Arabidopsis thaliana, Caenorhabditis elegans, Danio rerio, Drosophila melanogaster, Escherichia coli, Homo sapiens, Mus musculus, and Saccharomyces cerevisiae. The GEO database organizes raw samples of gene expression data in series and platforms, as described in Section 3.1. GENIUS includes a total of 133,343 samples of the Affymetrix microarray platform, coming from 7,073 series that cover a wide range of experimental conditions (Table 3.1). The table shows the accession of the GEO platforms (microarrays) included for each organism, as well as the number of series, samples, probes, genes, and features for each of them. In addition, the last column shows the percentage of genes mapped in the arrays that are annotated in the Biological Process (BP) branch of Gene Ontology (GO). We did not consider annotations with an IEA evidence to derive these statistics. Gene Ontology annotations for all supported organisms were downloaded from GO on February, 2015. GENIUS contains annotations from the three branches of GO: biological process, molecular function, and cellular component. For reference, Table 3.1 shows the percentage of matched genes that have at least one annotation in the biological process branch of GO. This percentage excludes annotations in the root term (GO:0008150) and annotations with an IEA evidence code. By default, GENIUS does not include annotations with an IEA evidence code, because these have not been manually curated. However, the user can choose which evidence codes to include when adding GO terms to the query list in GENIUS (Figure 4.7B).

Table 3.1. Statistics of the gene expression data included in GENIUS.



Organism	GEO Platform	Nr Series	Nr Samples	Nr Features
Arabidopsis thaliana	GPL198	742	8,872	12,614
Caenorhabditis elegans	GPL200	113	1,671	2,911
Danio rerio	GPL1319	107	1,340	1,216
Drosophila melanogaster	GPL1322	236	3,570	6,440
Escherichia coli	GPL3154, GPL199	187	1,971	4,869
Homo sapiens	GPL570	2,968	80,228	30,747
Mus musculus	GPL1261	2,564	33,360	26,311
Saccharomyces cerevisiae	GPL90	156	2,331	5,506

Organism	Nr Features	Nr Probes	Nr Genes	% Genes in GO BP
Arabidopsis thaliana	12,614	22,810	20,366	65 %
Caenorhabditis elegans	2,911	22,625	17,432	34 %
Danio rerio	1,216	15,617	10,102	16 %
Drosophila melanogaster	6,440	18,952	12,741	51 %
Escherichia coli	4,869	17,520	4,215	29 %
Homo sapiens	30,747	54,675	20,307	58 %
Mus musculus	26,311	45,101	20,966	53 %
Saccharomyces cerevisiae	5,506	9,335	6,080	74 %

3.5.3. Mapping Microarray Probes to Genes

For simplicity, previous sections describe the DLS algorithm in terms of samples of gene expression profiles. However, internally, DLS works over expression data coming from probes of microarrays. Also, it is more practical for the inputs and outputs of GENIUS to be expressed in terms of genes and their identifiers, which are commonly used by biologists in contrast to probes identifiers. Thus, we map probes to genes using the metadata provided by Affymetrix. During this mapping process, we address two common design issues that can lead to biased expression signatures and artifact predictions as follows. The first issue relates to the existence of probes that match multiple genes (1 to N relationship). The second issue relates to the existence of different probes that match a unique gene (N to 1 relationship). We include these probes by addressing three main problems:

i. Probes that match the same gene (1 to N) are likely to show similar expression profiles and thus, to form (uninteresting) links between them.
ii. Genes matched by multiple probes (N to 1) would have a greater overall weight when evaluating the Expression Signature Score (ESS) (Equation 3.1).
iii. Genes matched by multiple probes (N to 1) would appear represented by multiple nodes in the predicted network, one for each probe containing an expression signature.

To address issues 1 and 2 above we add an overall weight to the coexpression of each training probe. More specifically, when training a specific probe p that matches a gene g_i, the weight of any additional probe matching g_i is set to 0 (issue 1). Also, if a gene $g_j \neq g_i$ is matched by *N* different probes, then the weight of each of these *N* probes is set to 1/N (issue 2). In other words, the weights of probes matching a common gene sums 1 if $g_j \neq g_i$, and 0 otherwise. To tackle issue 3, all probes matching the same gene in the output network are merged into a unique node containing all their associations. More precisely, let $P(g_i)$ be the set of probes that match gene g_i. Then, an edge e_{ij} from the node of g_i to the node of g_j is inserted if and only if there exists a probe in $P(g_i)$ that predicts a probe in $P(g_j)$. Then, the attributes of the edge e_{ij}, like the confidence of the prediction or the coexpression level between g_i and g_j are extracted from the probe in $P(g_i)$ that predicts a probe in $P(g_j)$ with the highest confidence.

The original version of DLS simply discarded these problematic probes, and thus, only the ones having a 1 to 1 relationship with genes where considered (Puelma et al., 2012). However, the current implementation offers a better solution as it solves this problem for microarray designs that include a much larger number of problematic probes without losing valuable data.

3.5.4. Scores and graph theory indicators for gene prioritization

Gene networks, as most biological networks, usually have a hierarchical and scale-free topology, where most nodes have a small number of connections and only a few nodes are highly connected (hubs) (Barabási & Oltvai, 2004). Most networks predicted by GENIUS also show this kind of topology. This allows us to take advantage of network theory and use centrality indicators to pinpoint relevant genes for the biological function of interest (Azuaje, 2014). GENIUS allows users to easily do this from the "Genes" tab in the results screen. This tab contains a table that displays all the genes in the network and several properties that can be used to rank them. By default, GENIUS displays genes ranked by their overall centrality (OC), a custom score obtained by calculating the mean of the normalized values of the degree centrality (DC) and the betweenness centrality (BC) of nodes:

$$OC_i = \frac{DC_i/\max(\overrightarrow{DC}) + BC_i/\max(\overrightarrow{BC})}{2} \tag{3.10}$$

The degree DC_i measures the total number of nodes connected to a node i. Thus, this indicator is useful to highlight genes that are coexpressed with many other genes, acting as central hubs (Yu, Kim, Sprecher, Trifonov, & Gerstein, 2007). In contrast, the betweenness centrality BC_i measures the proportion of shortest paths between all pairs of nodes that pass through a node i. Thus, this indicator is useful to highlight genes that act as bottlenecks, making them essential for the flow of information and the overall connectivity of the network (Yu et al., 2007). Section 4.4.2 shows the usefulness of these centrality indicators to find key genes for drought tolerance in plants, which is a complex trait involving several biological processes.

In addition to the graph theory indicators explained above, GENIUS also provides various scores and properties, which serve as evidence to evaluate and prioritize the predicted genes according to how likely it is for them to be involved in the biological function of interest. More specifically, the "Predictions" tab of the results screen shows a summary of the coexpressions, confidences and a global score of each predicted gene, as well as the in-degree of their respective nodes (Figure 4.8). In GENIUS, all edges are directed and each edge represents a prediction. More precisely, an edge from a node A to a node X, with weight ω, represents that a positive gene A predicts a gene X as positive, with confidence ω (Details in Section 3.3.2). Consequently, the in-degree of a node X represents the number of positive genes that predict gene X as positive. In addition, the prediction score for gene X corresponds to the sum of all the confidences given to gene X. By default, GENIUS displays the predicted genes in this tab ranked by their prediction score. Finally, the table shows the coexpression and confidence for a predicted gene X, which correspond to the maximum values obtained among all genes predicting gene X.

In addition to display the results tables in the web application, GENIUS allows users to download them in excel format, so that they can make further analyses of the scores and indicators obtained in their prediction. Also, GENIUS allows users to export the resulting network to Cytoscape (Smoot et al., 2011), where power-users can make additional network analyses.

4. RESULTS AND DISCUSSION

This chapter shows and discusses the results of several evaluations of the DLS algorithm, as well as several applications of it, through the GENIUS web server, to research case studies. Section 4.1 presents the evaluation performed in the original publication of the DLS algorithm (Puelma et al., 2012). Section 4.2 presents a real research scenario in which DLS is successfully used to make a novel discovery that was later experimentally validated. Since the original publication of DLS, the algorithm has received some changes to further improve its performance. Section 4.3 presents an updated evaluation of the current DLS algorithm, which is described in detail in Chapter 3, and used in the GENIUS web server. These updated evaluations show an improvement in the prediction performance over the original DLS algorithm and thus, the insights and conclusions presented in Section 4.1 should hold for the implementation presented in this thesis. Finally, Section 4.4 presents evaluations and applications of the GENIUS web server through two representative case studies.

4.1. Evaluation of DLS using A. thaliana datasets from 2008-2010

In our original work where we presented DLS (Puelma et al., 2012), we performed several systematic evaluations in order to measure five aspects of the DLS algorithm:

i. Impact of selecting pairs of control-test conditions automatically.
ii. Impact of the number of experimental conditions available.
iii. Usefulness of the False-Negatives Discovery procedure.
iv. Performance comparison with state-of-the-art methods.
v. Ability to predict new annotations from GO.

In our tests, we use two expression datasets for the *A. thaliana* organism: one defined by an expert with 643 features (expert-dataset) and another defined by our automatic procedure described in Section 3.1.1 with 3911 features (automated-dataset). Also, we measured the impact of the number of experimental conditions available by extracting two additional smaller datasets from the automated-dataset, defined by a random selection 1000 and 2000 features, respectively. In addition, we measure the usefulness of the False-Negatives Discovery (FND) procedure described in Section 3.4 by using two configurations of DLS: one using the FND procedure (FND-DLS), and another without using it (DLS). Finally, we measure the ability of DLS to predict new genes by comparing its performance against two widely used state-of-the-art algorithms: SVMs (Brown et al., 2000) and CNs (Vandepoele et al., 2009). We measured the prediction performance of the different configurations over 101 biological processes from GO. For this, we first perform cross-validation tests using annotations from year 2008. Then, we measure the ability of the algorithms to predict new annotations from year 2010 using an enrichment analysis.

The rest of this section starts explaining the details of the experimental setup used for these evaluations. Then, it discusses the results obtained and the insights we extract from them.

4.1.1. Experimental setup

Gene expression datasets

In our tests we used two gene expression datasets. In the first dataset, pairs were manually defined by an expert, starting from a total of 2017 *Arabidopsis thaliana* ATH1 microarray slides (including replicates). Of these 2017 slides, 1907 were obtained from Nottingham Arabidopsis Stock Centre (NASC) (www.affymetrix.arabidopsis.info), 26 from Gene Expression Omnibus (GEO) repository (http://www.ncbi.nlm.nih.gov/geo/), and 84 from experiments done in our laboratory. This process produced a dataset with $M = 643$ features, which we refer to as the "expert-dataset". In the second dataset, pairs were derived by the automatic procedure described in Section 3.1.1, starting from an updated raw dataset containing 3352 microarray samples. The additional samples were extracted later from NASC. This process produced a data matrix with $M = 3911$ features which we refer to as the "automated-dataset".

Gene Ontology annotations

The evaluations consider the selection of 101 representative GO-terms from the 3500+ GO-terms available for *A. thaliana* in the biological process ontology. This selection was performed using the annotations available in GO on May 8, 2008. First, we filtered out all the annotations with IEA evidence code (Inferred from Electronic Annotation), as they are not reviewed by a curator and thus, are not reliable. Then, we selected representative functional GO-terms using a depth-first strategy, searching for the first GO-term of each branch containing between 30 and 500 annotated genes. The lower bound of 30 genes ensures for the positive sets to have a significant number of samples. The upper bound of 500 genes removes the GO-terms that are too broad and biologically less interesting. Nevertheless, this selection is representative of the space of possible biological processes in the sense that all the branches of the GO DAG are represented by at least one GO-term in our selection. In other words, all the GO-terms that were filtered out are either subcategories (descendants) or broader categories (ancestors) of at least one of the selected GO-terms. The selected GO-terms are Levels 2–6 in the GO hierarchy and cover a wide range of biological processes, such as responses to different stimulus and various metabolic and developmental processes. Also, we extracted new annotations available on GO on September 7, 2010 for each of the 101 selected GO-terms, in order to perform enrichment analyses for each method (Section 4.1.2).

Training sets

We derived a labeled training set for each of the 101 selected GO-term, as described in Section 3.1.2. However, the evaluations shown in section 4.1 use the method incorporated in the original published DLS algorithm (Puelma et al., 2012) to derive negative sets, in contrast to the Rocchio algorithm used in the latest implementation of DLS for GENIUS, and used in evaluations of sections 4.2 and 4.3 (Youngs et al., 2014). The number of positive genes in the 101 training sets varies from 30 to 474, with an average of 162 genes. The number of negative genes varies from 1011 to 4112, with an average of 3105 genes. Thus, in average, negative training sets contains 19.2 times more genes than the positive ones, which is expected, because most genes are not involved in a particular biological process.

Testing methods and metrics used to evaluate prediction performances

10-fold cross-validation tests were performed over each selected GO-term. Briefly, a 10-fold cross-validation consists of 10 trials done over 10 equally sized partitions (folds) of the training data. In each trial, a model is first trained using 9 of the 10 folds (90% of the samples are used as training set). Then, this model is used to classify the samples in the remaining fold (10% of the samples are used as testing sets). Thus, at the end of the 10 trials, each training sample has been classified once by an independently trained model. Then, the performance of a method can be evaluated by comparing the predicted labels with the known labels of the samples. In particular, in these tests we used the following metrics to evaluate performance:

$$\begin{aligned} Precision &= \frac{|TP|}{|TP|+|FP|}, \\ Recall &= \frac{|TP|}{|TP|+|FN|}, \qquad (4.1) \\ F_{\beta} - score &= \frac{1+\beta^2}{\frac{1}{recall}+\frac{\beta^2}{precision}}, \end{aligned}$$

where $|TP|$, $|FP|$ and $|FN|$ correspond to the number of true positives, false positives and false negatives, respectively. Precision measures the proportion of positive predictions that are correct. For example, a precision of 0.5 means that for every two positive predictions one is correct. Recall measures the proportion of positive genes that are predicted as positive. For example, a recall of 0.5 means that from the total number of positive genes half are predicted as positive. In other words, precision is as a measure of exactness or fidelity, whereas recall is a measure of completeness. Ideally, a method should obtain a high precision and a high recall, however, in practice there is generally a trade-off between these evaluation metrics. For this, we use the $F_{\beta} - score$, which provides a joint evaluation of both precision and recall by calculating their harmonic mean. It ranges between 0 and 1, being close to 1 when both, precision and recall, take values near 1, but close to 0 when any of these two metrics is close to 0. The β parameter controls the weight given to precision with respect to recall. In our tests, we used β = 2 ($F_2 - score$), in order to favor accurate models over models with high recalls but large false positive rates.

An inconvenience of the cross-validation analysis is its sensibility to the presence of false negatives, this is, when positive genes are incorrectly labeled as negatives, which can affect both precision and recall metrics. On one side, false negatives present in the testing sets can make some positive predictions to be incorrectly interpreted as erroneous predictions (false positives), which will increase the actual false positives rate and, in consequence, will lower the real precision of the algorithm. On the other side, false negatives present in the training sets can diminish the ability of the algorithm to find discriminative patterns, decreasing the number of positive predictions (i.e. $|TP| + |FN|$) and, in consequence, lowering the recall of the algorithm.

In order to reduce the prejudicial effects of false negatives in the results and to perform a more realistic evaluation, we used an enrichment analysis of new annotations. Briefly, the analysis consisted in training the methods using the annotations from year 2008 and then analyze the positive predictions to test how enriched they were in new annotations that became available in September 7, 2010 (Figure 4.1). The enrichments were tested using a hypergeometric distribution and a P-value threshold of 0.1 to consider enrichment. Given a set of N_T total samples, of which N_P are positive, this distribution measures the probability (*P-value*) of drawing M_P positive samples by chance, when drawing a total of M_T samples. Then, if the P-value is lower than a predefined threshold, the subset of M_T samples is said to be enriched in positive samples. In our evaluation, N_T corresponds to the total number of unlabeled genes (i.e. the genes that will be classified), according to the training sets derived from year 2008 annotations. N_P corresponds to the number of unlabeled genes that are actually positive according to the new annotations from year 2010. M_T corresponds to the number of genes predicted as positives. Finally, M_P correspond to the number of genes correctly predicted as positives.

In order to facilitate the analysis of the results, we summarized them using three criteria. The first criterion consists of counting the number of GO-terms in which each method attains useful predictions. In the case of cross-validations, we consider as useful the predictions in GO-terms with *precisions* greater than 0.33 (i.e. at least one of three predictions must be correct) (Figure 4.1A). In the case of the enrichment analyses, we consider as useful the predictions for GO-terms that obtain a *P-value* lower than 0.1 (i.e. enriched predictions) (Figure 4.2A). The second criterion consists of evaluating the average performances of the methods considering all the 101 tested GO-terms. In cross-validations, we include *precision*, *recall* and F_2*-score* averages (Figures 4.2B–D). In the case of enrichment analyses, we include *P-value* averages (Figure 4.3). Finally, the third criterion consists of a pairwise comparison of the performances of the methods over each GO-term. Given two methods, A and B, we counted the number of useful GO-terms in which A outperforms B and vice versa. We used the F_2*-scores* and *P-values* as performance measures for cross-validations (Figure 4.2E) and enrichment analyses (Figure 4.3C and Figure 4.4), respectively.

Workflow used for evaluations

First, we performed cross-validation and enrichment analyses using the expert-dataset as described in the text above. The automated-dataset is evidently more prone to both useless and redundant features, as some of them may be defined using biologically meaningless comparisons. Thus, the expert-dataset was used in order to ensure quality control–test condition pairs for the evaluations. In addition, we performed enrichment analyses using the automated-dataset with two specific aims: (i) test the potential of the automated-dataset for function prediction and (ii) test the performance of the methods in datasets with an increasing number of conditions (features). Thus, we performed enrichment analyses in two additional smaller datasets, defined by a random selection of 1000 and 2000 features from the automated-dataset.

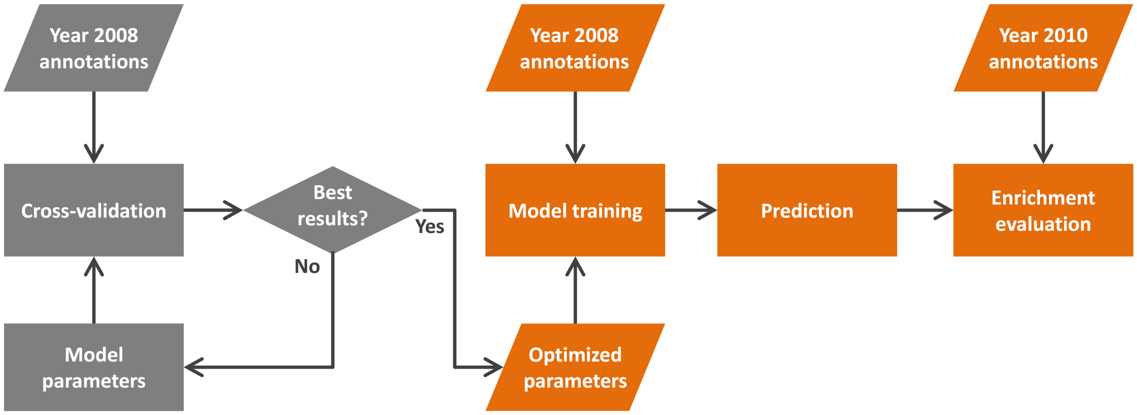


Figure 4.1. Strategy to optimize and test the prediction performance of methods.

Evaluated methods and configurations

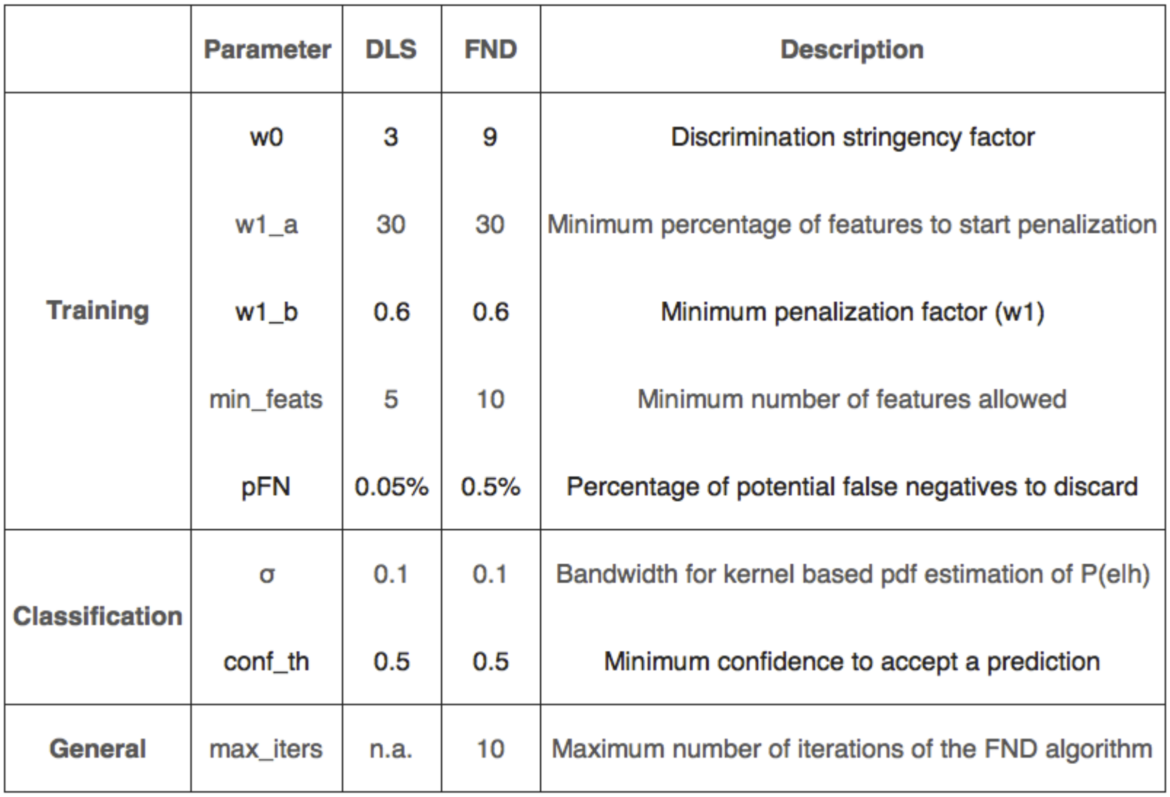
In terms of the evaluated methods, we used the configuration and parameters providing the highest average F_2*-score* in the cross-validation analysis (Figure 4.1).

For DLS, we used two alternative configurations, one using the False Negative Discovery method (FND-DLS), described in Section 3.4 and other without using it (DLS).

To select the final parameters for DLS (Table 4.1), we first fixed some of them and then optimized the remaining ones. In the training phase, we fixed the minimum number of features of signatures (*min_feats*) to five. This parameter should not be relevant, because the automatic penalization in the feature selection algorithm should prevent finding small signatures. Also, during the classification phase, we fixed the minimum confidence (*conf_th*) to a conservative threshold of 0.5. Using this threshold, a gene is predicted as positive whenever the method has more confidence about the gene being positive than negative. Although better performances could be obtained by optimizing this parameter, we used this criterion to be fair in the comparison with SVMs, which uses this criterion to classify the samples. As for the optimized parameters, we obtained the best results by giving a weight equal to three to negative genes (w_0 = 3 in Equation 3.1), which allows the method to search for subsets of conditions that are highly discriminative. Also, we obtained the best results by activating the option to discard the top 0.05% most coexpressed negative genes (*pFN*) during the training process. This allows the method to prevent overfitting the false negatives in the training set. Although this option slightly reduced the *precision*, it helped to improve the *recall* in a greater extent, resulting in an overall higher F_2-*score*.

As for the parameters of the FND method in DLS-FND, we used a maximum of 10 iterations (*max_iters*), and a *pFN* equal to 0.5% instead of 0.05%, in order to increase the number of potential false negatives that it can find during its training phase. Also, we used the option to move the predicted FNs from the negative to the positive training set after each iteration. Thus, after the execution of the FND method, DLS performs the final prediction over a training set with a larger number of positive genes and a lower number of negative genes than the original training set.

Table 4.1. Optimized parameter values for DLS and FND.



	Parameter	DLS	FND	Description
Training	w0	3	9	Discrimination stringency factor
	w1_a	30	30	Minimum percentage of features to start penalization
	w1_b	0.6	0.6	Minimum penalization factor (w1)
	min_feats	5	10	Minimum number of features allowed
	pFN	0.05%	0.5%	Percentage of potential false negatives to discard
Classification	σ	0.1	0.1	Bandwidth for kernel based pdf estimation of P(elh)
	conf_th	0.5	0.5	Minimum confidence to accept a prediction
General	max_iters	n.a.	10	Maximum number of iterations of the FND algorithm

For SVM, we used the C-SVC implementation of SVM from the LIBSVM library (Chang & Lin, 2001) available for MATALB®. In order to select the kernels, we evaluated four types of kernels: radial basis (RBF), linear, and two polynomial kernels with degrees equal to two and three, respectively. In line with results previously reported (Brown et al., 2000), RBF-SVM shows the best performance. However, as linear-SVM has the advantage of being more easily interpreted, it provides a good alternative and reference point to compare the performance of our method. Consequently, we report the results of both, RBF-SVM and linear-SVM. In terms of the selection of relevant parameters for the different SVM models, we tested different configurations following the default values suggested by the LIBSVM library and the optimization methods proposed by Brown et al., 2000. The linear and polynomial kernels follow the function:

$$K(u, v) = (gamma * u' * v + coef0)^{degree} \tag{4.2}$$

with parameter values specified in Table 4.2.

Table 4.2. Linear and Polynomial SVM parameters.

	Linear	Polynomial 2	Polynomial 3
$degree$	1	2	3
$coef0$	1	1	1
$gamma$	$1/M$	$1/M$	$1/M$

M corresponds to the number of features in the dataset. The value of *coef0* was selected according to what is reported by Brown et al., 2000. In addition, the RBF kernel follows the function:

$$K(u, v) = exp(-gamma * |u - v|^2) \tag{4.2}$$

We used the default value provided by the library for gamma (1/*M*) in all kernels. Also, we used equal weights for positive and negative samples.

For CN, we constructed the networks using the cosine correlation metric to define coexpression associations. Predictions were performed using a guilt-by-association criterion over the neighbors of each gene, using the hypergeometric distribution and Bonferroni correction for multiple tests, as described by Vandepoele et al., 2009. We tested networks with five different correlation thresholds: 0.5, 0.6, 0.7, 0.8 and 0.9. In addition, we tested three P-value thresholds for the hypergeometric distribution: 0.1, 0.05 and 0.01. Here, we report the results of the CN model using a correlation and *P-value* thresholds of 0.6 and 0.1, respectively, which provided the highest average *F_2-score* in the cross-validation analyses.

4.1.2. Results and discussion

The results of the cross-validation and enrichment analysis using the expert-matrix are summarized in Figures 4.2 and 4.3, respectively. The results of the enrichment analysis using the automated-matrix are summarized in Figure 4.4.

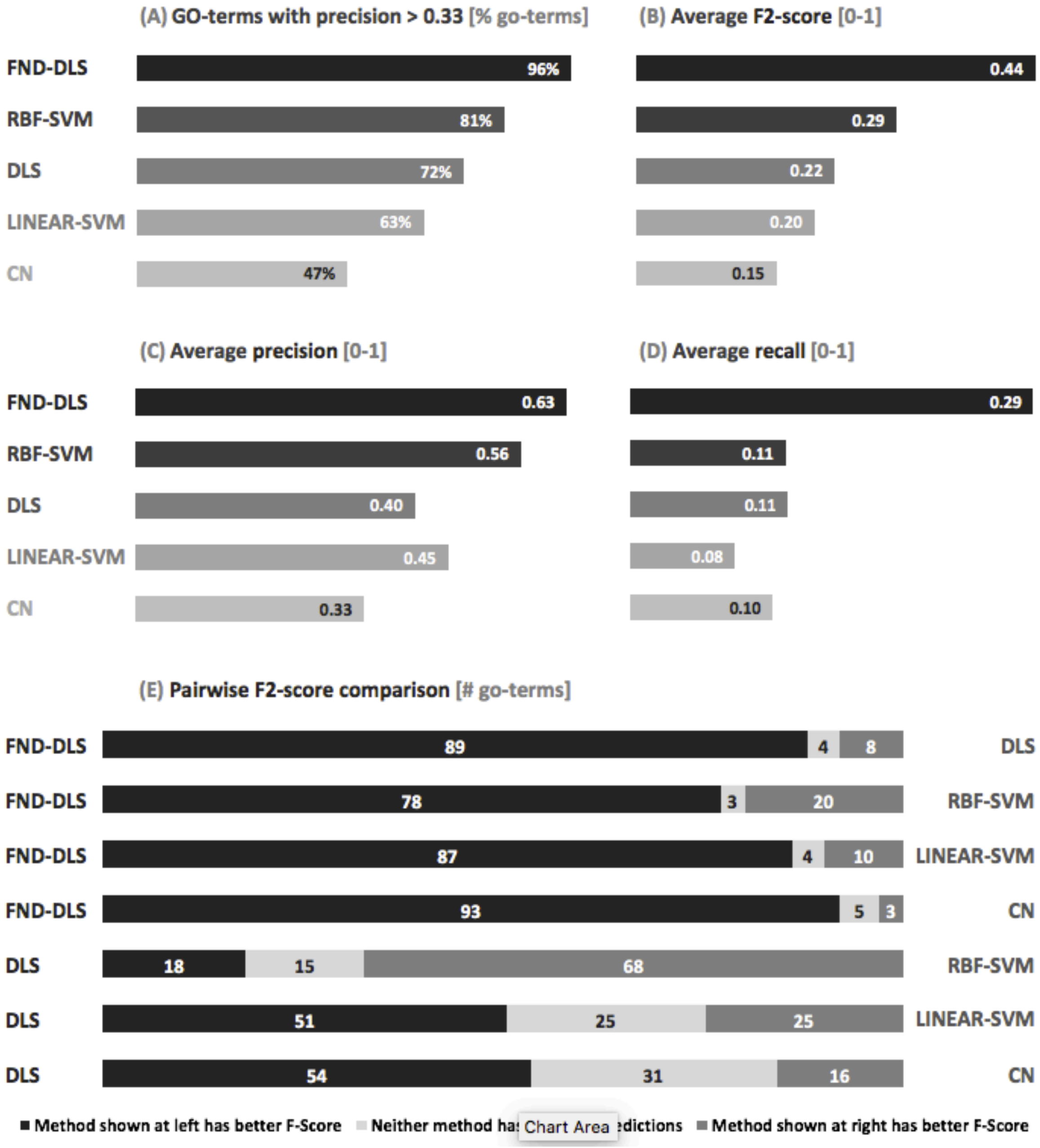


Figure 4.2. Results of the 10-fold cross-validation analyses.

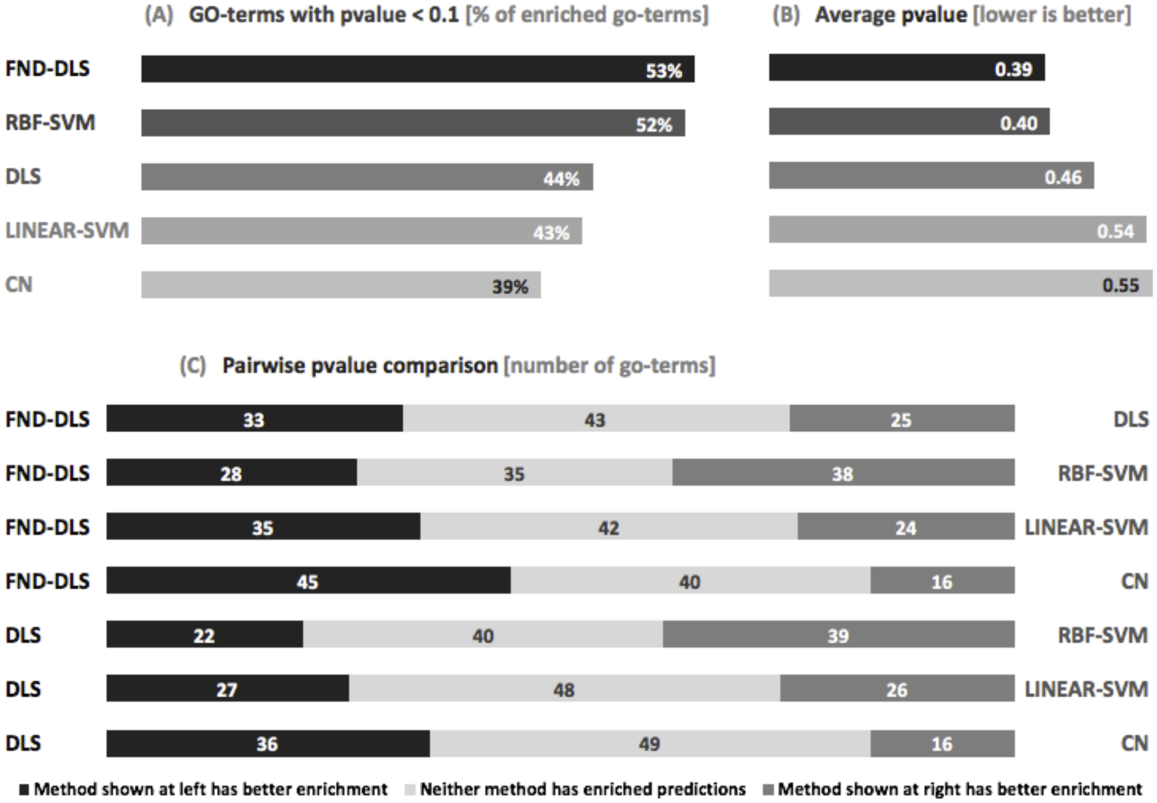


Figure 4.3. Pairwise comparison of the results of the enrichment analyses.

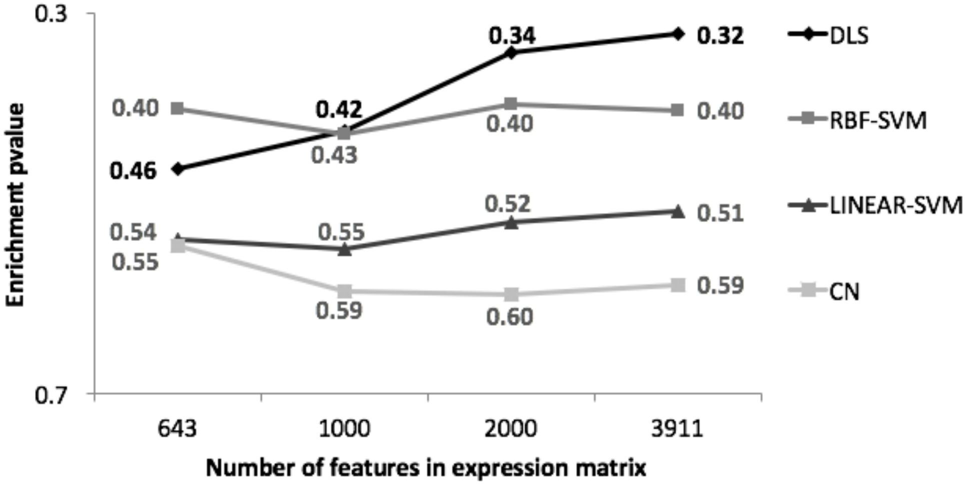


Figure 4.4. Results of the enrichment analyses performed over datasets with increasing number of features.

FND-DLS shows the best overall prediction performance

Our results show that FND-DLS outperforms all competing methods, whereas RBF-SVM consistently attains the second best performance. In the case of cross-validation analyses, FND-DLS attains useful predictions (*precision* > 0.33) in 96% of the considered GO-terms, corresponding to 15% more GO-terms than RBF-SVM, 24% more than DLS, 33% more than linear-SVM and 49% more than CN (Figure 4.2A). In addition, it attains an average F_2-*score* of 0.44, whereas RBF-SVM, DLS, linear-SVM and CN attain averages equal to 0.29, 0.22, 0.20 and 0.15, respectively (Figure 4.2B). Although FND-DLS attains better average *precisions* than the other methods (Figure 4.2C), its supremacy in terms of the F_2-*score* is mostly explained by its higher recalls. FND-DLS attains an average *recall* of 0.29, whereas RBF-SVM, DLS, linear-SVM and CN attain average recalls equal to 0.11, 0.11, 0.08 and 0.10, respectively (Figure 4.2D).

The overall small *recall* levels obtained by the methods may be explained by four main factors:

i. Some genes may be regulated under experimental conditions not available in the expression dataset.
ii. Some genes are not regulated at a transcriptional level and thus, may not have (common) expression patterns.
iii. Due to missing functional labels, some genes may be regulated by (or regulate) genes that are not present in the positive training, which makes impossible for the methods to discriminate them.
iv. Genes with false negative labels may share and mask some discriminative patterns present among positive genes.

The higher recall levels achieved by FND-DLS over the other methods remark the importance of the last two factors described above.

The higher precisions obtained by FND-DLS supports the effectiveness of the FND process. The FND process iteratively moves the predicted FNs to the positive set.

Thus, if FND predicted FNs incorrectly, these genes would become false positive genes which, in turn, would decrease the precision of FND-DLS.

Cross-validation is useful to assess the relative performance of the methods, however, its results must be considered with caution (Varma & Simon, 2006). To tackle this, we use an enrichment analysis over a completely new set of labeled genes (i.e. new annotations from year 2010) to assess the performance of our method in an alternative and more realistic scenario (details in Section 4.1.1).

Remarkably, the results of the enrichment analysis confirm the supremacy of FND-DLS over RBF-SVM, although its overall advantage is smaller than in the cross-validation test (Figure 4.3). FND-DLS attains enriched predictions in 53% of the GO-terms, whereas RBF-SVM attains enriched predictions in 53% of them, DLS in 44%, linear-SVM in 43% and CN in 39% of them (Figure 4.3). Note that some GO-terms have few or no new genes annotated on year 2010 with respect to year 2008 and thus, it is very difficult or even impossible for the predictions to be enriched. In addition, the enrichment performance is affected by the same four factors exposed above for cross-validation. In terms of enrichment *P-value* (lower *P-value* represent higher enrichments), FND-DLS attains an average of 0.39, whereas RBF-SVM attains an average of 0.40, DLS of 0.46, linear-SVM of 0.54, and CN of 0.55 (Figure 4.3B).

Discriminative methods, DLS and SVM, provide more accurate gene function predictions than CNs

According to our experiments, both versions of DLS and SVM outperform CN. Although CN obtains similar average *recall* levels than SVMs and DLS (without FND), it fails in providing predictions as precisely as them (Figure 4.2C and D). These results show the advantages of using discriminative training techniques in contrast to semi-supervised techniques in attaining accurate gene functional predictions. This assertion is further supported by the results of the enrichment analysis.

There is no method to rule them all

Although FND-DLS and RBF-SVM show the best overall performances, when comparing the performance at a term-by-term scale, we can only conclude that there is no method able to attain the best performance through all GO-terms (Figures 4.2E and 4.3C). There are many factors that can bias the predictability of genes of a biological process toward one method or another. For example, in GO-terms related to responses, we see a bias in the predictability toward DLS in expense of CN, as the responses are usually expressed under specific environmental or physiological conditions, which DLS is able to detect due to its local search for discriminative features.

The discriminative and local expression patterns of DLS provide effective and meaningful predictions

According to the FDR matrix X_{FDR}, 96.2% of the expression changes in the log-ratio matrix X_{LR} are not significant in the expert-dataset (considering an $FDR<0.1$ for significance). This means that on average, genes show differential expression in only 24 (3.8%) of the 643 features. This sparseness emphasizes the importance of selecting relevant features to achieve effective predictions. SVM perform transformations to higher dimensions, which can also be interpreted as an implicit selection of relevant features. However, in the case of non-linear SVMs, these transformations complicate the interpretation of the predictions and the extraction of further knowledge. Consequently, besides the prediction power of DLS, a key advantage over non-linear SVMs and other discriminative state of the art prediction methods is its ability to provide biologically meaningful and interpretable predictions, while maintaining highly accurate predictions. Unlike SVMs, DLS is able to visually expose its predictions in the form of a network. This network delivers a much richer interpretability to the user than SVMs, providing key information about the regulatory linkages that may exist between the genes of the biological function of interest. Finally, unlike both SVMs and CNs, DLS is able to explicitly reveal the experimental conditions and genes that are relevant for each prediction, by exposing the features and genes that define each expression signature.

DLS systematically improves its performance as more experimental conditions are added to the dataset

As stated above, the lack of informative features is one of the factors that may affect the prediction potential of the methods. In this sense, the increasing amount and variety of gene expression experiments represent both an opportunity and a challenge. If the number of available experiments increases, chances for it to have more informative features increase. However, the amount of uninformative and redundant features should also increase, adding extra noise that must be correctly handled by the prediction methods.

The results of the enrichment analyses performed using the automated-dataset support our previous hypothesis and one of the most remarkable features of DLS (Figure 4.4). When using the expert-dataset, containing 643 features, DLS achieves an overall *P-value* of 0.46. Interestingly, when using the automated-datasets, containing 1000, 2000 and 3911 features, its average *P-value* improves to 0.42, 0.34 and 0.32, respectively. In contrast, RBF-SVM is not able to improve its performance, linear-SVM shows little improvement, and even more, the performance of CN worsens. Notice that when using the automated-dataset, DLS achieves the highest overall performance in terms of enrichment, even without using the FND procedure (Figure 4.4).

These results show that DLS is able to overcome the underlying noise added by the automated-dataset by effectively extracting relevant and informative features. In addition, they support the usefulness of our automatic procedure to generate log-ratio expression datasets from poorly annotated experiments. But, most remarkably, they suggest that DLS should be the most benefited method as, in the future, more microarray experimental data becomes available. As the dataset becomes larger, the ratio of noisy vs useful features increases, which also increases the complexity of finding useful, discriminative patterns. This effect negatively impacts the performance of SVMs, especially linear-SVMs, and CNs. In contrast, the local modelling of DLS through the local search of expression signatures provides a more scalable approach to tackle the added noise while exploiting the added informative features.

4.2. Application of DLS to search key genes regulating nitrogen use efficiency of plants

In the previous section, we showed the capabilities of DLS to predict new genes participating in a biological function of interest by performing a systematic evaluation. In this section, we show the capabilities of DLS to predict informative and biologically coherent gene networks that can pinpoint key genes modulating complex biological functions or traits of interest. In order to show these capabilities, we show how DLS was used in a real research scenario to find key genes modulating nitrogen use efficiency (NUE) in plants (Araus et al., 2016).

Improving nitrogen use efficiency of plants is key to tackle serious problems in today's agriculture, food production and ecology. Nitrogen (N) is an essential macronutrient and a key element controlling plant growth, development and productivity. Use of N-based fertilizers has increased more than 8 fold in the last 50 years to cope with increasing demands of agriculture and food production (Dawson & Hilton, 2011). Intensive use of N-fertilizers is causing major detrimental impact on the ecosystem, including eutrophication of waters and increase of gaseous emissions of toxic N oxides and ammonia to the atmosphere (Ju et al., 2009; Lassaletta et al., 2014; Robertson & Vitousek, 2009). Moreover, excessive use of fertilizers is a major cost for farmers, which in turn affects the commercial price of vegetables and fruits. In this context, it is of paramount importance to design strategies to improve NUE for increased plant productivity in sustainable and environmentally responsible ways (Gutierrez, 2012).

Many efforts have been devoted towards defining target genes for generating crops with enhanced NUE (Crawford & Forde, 2002). However, NUE is a complex genetic trait, encompassing multiple metabolic, physiological, and developmental processes in plants exposed to a changing environment. Due to this complexity, it is very difficult for biologist to know which of the hundred genes involved in these processes can be good candidates to impact NUE. In addition, experimentally testing NUE is a time-consuming process, which involves growing plants under different nitrate concentrations and then measuring the total seed amount produced per plant. Thus, it is crucial to identify sound candidate genes in which to focus experimental tests, for which we used DLS.

One of the most useful DLS outputs is a discriminative gene network (Section 3.3.2), which can be analyzed using standard network topology statistics and tools to pinpoint key genes for the regulation of a biological function of interest (Azuaje, 2014) (Section 3.5.4). Given that NUE is a complex process that integrates various biological processes, we defined a positive set by including the genes annotated in 12 biological processes that are known to impact or control NUE in plants. Six of these processes are associated with N metabolism (nitrate assimilation GO:0042128, nitrate transport GO:0015706, ammonium transport GO:0015696, ammonium response GO:0060359, nitrogen response GO:0019740, nitrate response GO:0010167), while the other six are associated with plant development (regulation of seed development GO:0080050, organ senescence GO:0010260, endosperm development GO:0009960, vegetative to reproductive phase transition of meristem GO:0010228, vegetative phase change GO:0010050, seed maturation GO:0010431). The union of all these GO-terms resulted in a positive set with 220 genes. In order to do the prediction, we used the DLS configuration showing the best prediction performance in the enrichment evaluation presented in Section 4.1 (Figures 4.3 and 4.4). This configuration corresponds to FND-DLS, using GO annotations from September 7, 2010 and the automated-dataset containing 3,911 features for *Arabidopsis thaliana*.

Using this configuration, DLS predicted a network containing 350 genes. We analyzed this network using Cytoscape (Lopes C et al, 2010). First, we used the "Network Analyzer" plugin to calculate the degree and betweenness centrality of each node, two metrics that are commonly used to define nodes that are important for network structure (Azuaje, 2014). Then, we generated a network view in which nodes have sizes proportional to their degrees and colors according to their betweenness centrality (BC) values. For practical reasons, we extracted a subnetwork containing the 50 nodes with greatest degree (Figure 4.5). This network clearly highlights *BT2* as the most important gene for the overall network structure and topology. *BT2* is the node with the highest degree and betweenness centrality, making it the best candidate for experimental validation of its role in controlling NUE.

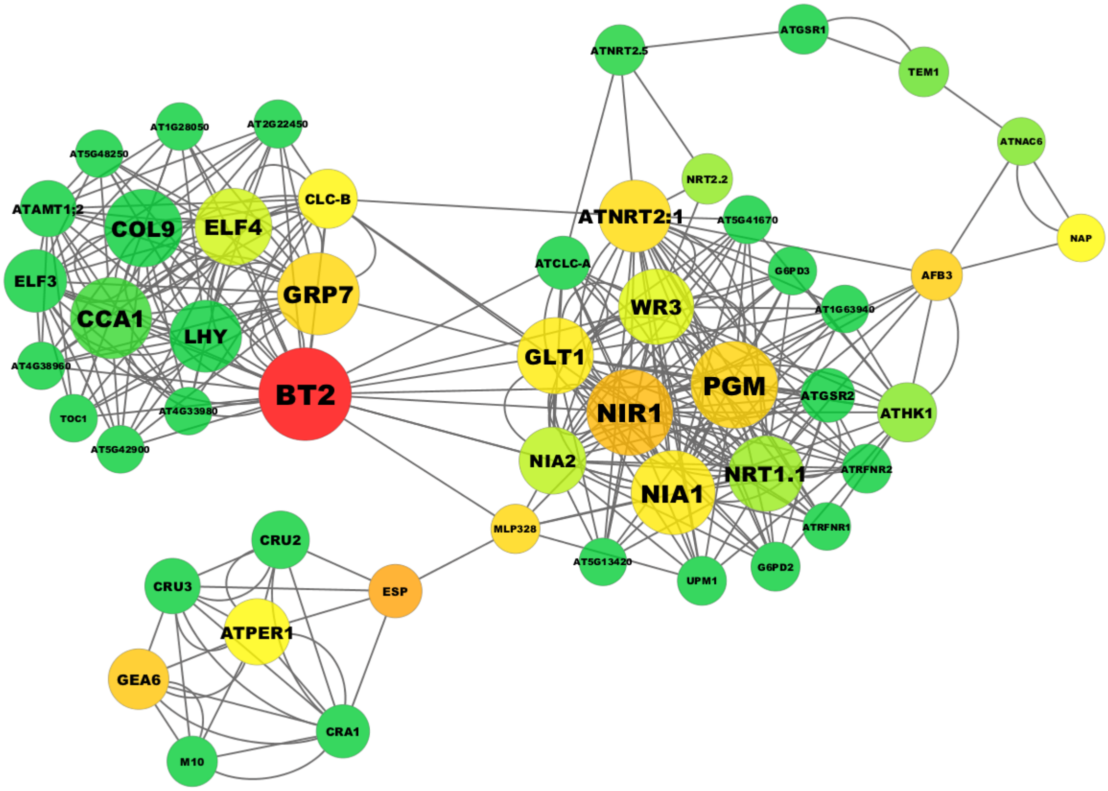


Figure 4.5. Network predicted by GENIUS for nitrogen use efficiency. BT2 is the node with greatest degree (the biggest) and betweenness centrality (the reddest).

In addition to *BT2*, we also tested its homolog *BT1*. BT2 belongs to a family of BTB and TAZ DOMAIN proteins composed of five members (Robert, Quint, Brand, Vivian-Smith, & Offringa, 2009) with BT1 (At5g63160) being the closest homolog with 80% sequence identity (L. Du & Poovaiah, 2004). Previous studies demonstrated BT1 and BT2 have functional redundancy and reciprocal transcriptional control during gametophyte development (Robert et al., 2009). Therefore, both BT1 and BT2 were selected for experimental validation.

The experiments performed in our lab found that under low nitrate conditions, NUE decreases in BT2 overexpressing Arabidopsis plants and increases in *bt1/bt2* double mutant *Arabidopsis* plants, as compared to control wild-type plants (Araus et al., 2016). Moreover, mutating the *BT1/BT2* ortholog gene in rice, *OsBT1*, increases NUE by 20% as compared to wild-type rice plants (Araus et al., 2016).

These results show the power of DLS to predict networks that highlight key genes for complex traits, a difficult and relevant challenge for many biological researches. They also show that the degree and the betweenness centrality are useful indicators to rank genes in the predicted networks and to pinpoint key genes modulating a biological function of interest. Based on these results, and in order to make it easier for biologist to perform this kind of analysis, we developed the GENIUS web server and included these indicators as part of its prediction results.

4.3. Evaluation of GENIUS-DLS predictions in an updated dataset from 2015

This section presents an evaluation of the current (2015) version of the DLS algorithm, annotations, datasets, and organisms included in the GENIUS web server (http://networks.bio.puc.cl/genius). In order to evaluate the prediction capabilities of GENIUS in all supported organisms and in a broad functional space, we performed an enrichment analysis over sets of representative GO-terms for each organism.

This evaluation is similar to the one performed for the former DLS implementation (Section 4.1, Figures 4.3 and 4.4), but differs in how we obtain the two sets of annotations needed to test enrichment. In our former evaluation (Section 4.1) we used an older annotation set (from year 2008) to obtain representative GO-terms and make predictions for each of them. Then, we used a newer annotation set (from year 2010) to test the enrichment of correct predictions. In contrast, in this evaluation we use a holdout strategy to obtain both annotation sets from annotations from year 2015, which are the ones included in GENIUS (Section 3.5.2). The holdout strategy consists of hiding (holding out) a random 30% of the genes annotated in each GO-term. In other words, this process produces an annotation set that artificially simulates an older annotation set having 30%

less annotations than the original set. Then, we use these two annotations sets to perform enrichment analyses in an analogous way as we did in our original evaluation (Section 4.1). In order to reduce the bias that can be produced by the random selection, we repeated the analysis over three independent, randomly generated holdout sets, registering the geometric mean of the enrichment *P-value* obtained over the three iterations for each representative GO-term. Analogously to our previous analysis, the enrichment *P-value* represents the probability of predicting a certain proportion of true positive samples by chance, given the total proportion of positive samples in the holdout set. Then, we consider GENIUS predictions for a GO-term to be enriched in correct predictions (true positives) if the final averaged *P-value* is lower than 0.1.

The holdout strategy used here has several advantages over the rollback strategy used in our former evaluation. In the rollback strategy, some GO-terms had many new annotations, while other had few or no new annotations at all, biasing the results over some GO-terms and making difficult or even impossible to obtain enriched annotations in other GO-terms. This bias would be even worst in the current scenario, where we have different organisms with different number of annotations. The current strategy diminishes this bias by ensuring that all tested GO-terms will have a 30% of “new” (held out) annotations to test enrichment. In addition, this strategy better covers the whole space of annotations by doing three analyses over independent holdout sets. Finally, it has the advantage of using the current annotation set instead of an older one, which allows it to better represent the current prediction performance.

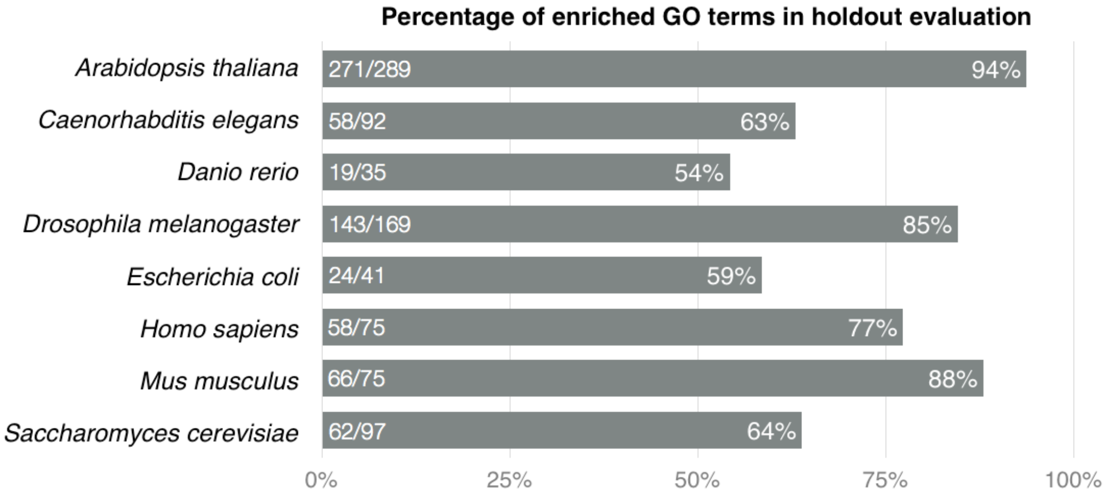


Figure 4.6. Prediction performance in supported organisms.

In order to summarize the results, for each organism we show the proportion of representative GO-terms in which GENIUS obtains predictions enriched in true positive genes (Figure 4.6). These results indicate that GENIUS can make high quality predictions in most GO-terms for the eight supported organisms. The best results are achieved for *A. thaliana* (94%), *M. musculus* (88%), *D. melanogaster* (85%), and *H. sapiens* (77%), while the worst results are obtained for *S. cerevisiae* (64%), *C. elegans* (63%), *E. coli* (59%) and *D. rerio* (54%) (Figure 4.6). Interestingly, these results correlate with the number of features (correlation of 0.59), as well as with the percentage of genes with GO annotation in the biological process branch (correlation of 0.62), available for each organism (Table 3.1). The correlation is even higher when considering both factors together (correlation of 0.71). The high correlation with the number of features is consistent with our previous results (Figure 4.4), which indicate that predictions should improve as more expression data becomes available (Section 4.1.2) (Puelma et al., 2012). The more expression data available, the higher the possibilities for DLS to find discriminative experimental conditions for each biological process and thus, to make more precise predictions. In addition, the high correlation with the number of annotations suggest that predictions should also improve as more annotations become available. Sparse annotations prevent the construction of informative training sets, increasing the number of false negative genes and lowering the number of true positive genes in them.

Clearly, this evaluation and the former evaluation are not completely comparable, because the strategies, datasets and implementations used are different in each of them. Nevertheless, there is a remarkable difference in the percentage of enriched GO-terms for *A. thaliana* in the current and former evaluations (94% (Figure 4.6) and 53% (Figure 4.3A) respectively), which suggest that GENIUS, with its updated DLS implementation and datasets, should provide a better prediction performance than the original version of DLS.

4.4. Applications of GENIUS web server to representative case studies.

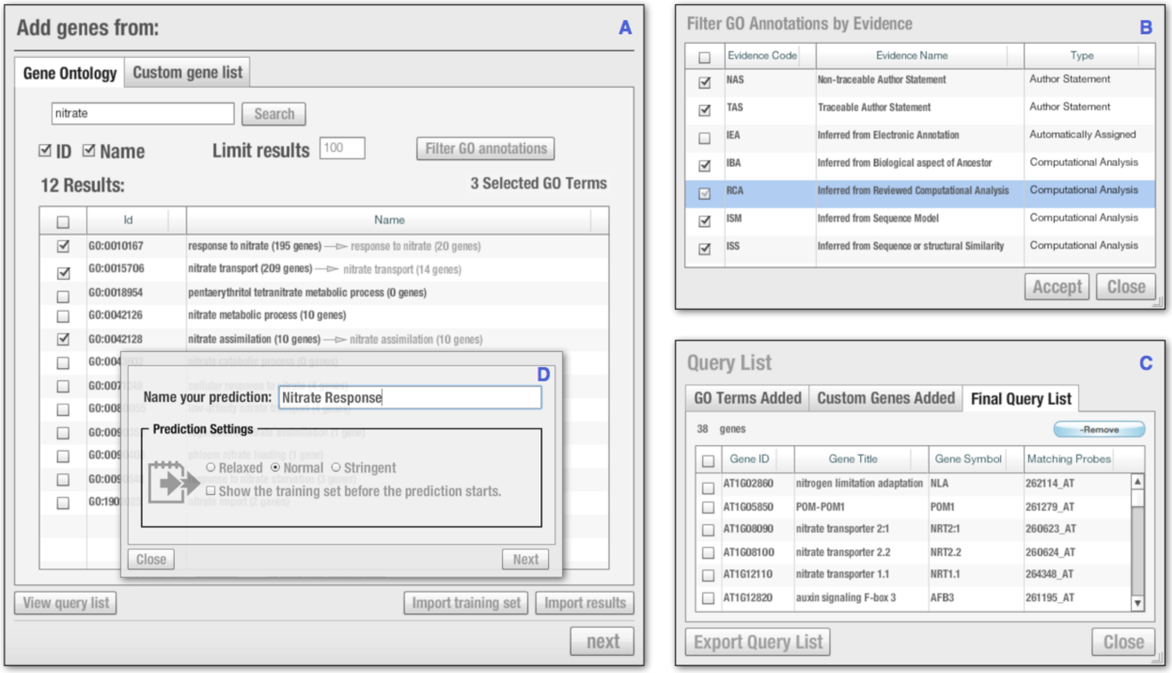
This section shows how GENIUS can be successfully used to guide research by showing two representative case studies. On both cases, we evaluate the predictions of GENIUS by searching the literature and by comparing them with state-of-the art application GeneMANIA (Mostafavi et al., 2008; Zuberi et al., 2013).

4.4.1. Predicting new genes for a biological function of interest

This section illustrates the use of GENIUS to predict new genes related to a biological function of interest (BF). For the purpose of this demonstration, we study the “nitrate response” in *Arabidopsis thaliana*. Nitrogen is a key macronutrient, essential component of amino acids, nucleic acids, pigments, hormones and many other biomolecules, and a major factor limiting plant growth and development. Nitrate is the main source of nitrogen for plants, and a signal that regulates global gene expression, physiology, and many growth and developmental processes (Gutierrez, 2012). Despite the fact that several transcriptome studies characterizing nitrate response are published (Vidal, Álvarez, Moyano, & Gutiérrez, 2015), there is still limited understanding about the regulatory factors and molecular mechanisms implicated in nitrate responses (Canales, Moyano, Villarroel, & Gutierrez, 2014).

Defining the query list and starting a new prediction

The first step to predict new genes involved in BF is to define a query list containing genes that are known to be involved in BF. For this, GENIUS offers users two complementing possibilities. The first is to directly add a list of gene identifiers. The second is to select a list of GO-terms related to the biological function of interest (Figure 4.7A). In the latter case, GENIUS adds to the query list all the genes annotated in these selected GO-terms.



Add genes from:

Gene Ontology | Custom gene list

nitrate

☑ ID ☑ Name Limit results 100

12 Results: 3 Selected GO Terms

☐	Id	Name
☑	GO:0010167	response to nitrate (195 genes) —▻ response to nitrate (20 genes)
☑	GO:0015706	nitrate transport (209 genes) —▻ nitrate transport (14 genes)
☐	GO:0018954	pentaerythritol tetranitrate metabolic process (0 genes)
☐	GO:0042126	nitrate metabolic process (10 genes)
☑	GO:0042128	nitrate assimilation (10 genes) —▻ nitrate assimilation (10 genes)
☐	GO:0043602	nitrate catabolic process (0 genes)
☐	GO:0071249	cellular response to nitrate (4 genes)
☐	GO:0080055	low-affinity nitrate transport (4 genes)
☐	GO:0090352	regulation of nitrate assimilation (1 gene)
☐	GO:0090408	phloem nitrate loading (1 gene)
☐	GO:0090548	response to nitrate starvation (3 genes)
☐	GO:1902025	nitrate import (2 genes)

Filter GO Annotations by Evidence

☐	Evidence Code	Evidence Name	Type
☑	NAS	Non-traceable Author Statement	Author Statement
☑	TAS	Traceable Author Statement	Author Statement
☐	IEA	Inferred from Electronic Annotation	Automatically Assigned
☑	IBA	Inferred from Biological aspect of Ancestor	Computational Analysis
☑	RCA	Inferred from Reviewed Computational Analysis	Computational Analysis
☑	ISM	Inferred from Sequence Model	Computational Analysis
☑	ISS	Inferred from Sequence or structural Similarity	Computational Analysis

Query List

GO Terms Added | Custom Genes Added | Final Query List

38 genes

☐	Gene ID	Gene Title	Gene Symbol	Matching Probes
☐	AT1G02860	nitrogen limitation adaptation	NLA	262114_AT
☐	AT1G05850	POM-POM1	POM1	261279_AT
☐	AT1G08090	nitrate transporter 2:1	NRT2:1	260623_AT
☐	AT1G08100	nitrate transporter 2.2	NRT2.2	260624_AT
☐	AT1G12110	nitrate transporter 1.1	NRT1.1	264348_AT
☐	AT1G12820	auxin signaling F-box 3	AFB3	261195_AT

Name your prediction: Nitrate Response

Prediction Settings

○ Relaxed ● Normal ○ Stringent

☐ Show the training set before the prediction starts.

Figure 4.7. Starting a prediction with GENIUS.

In our case study, we define the query list by selecting GO-terms related to nitrate response. For this, we search for GO terms with the "nitrate" keyword in the provided search box (Figure 4.7A). From the list of matched GO terms, we select three biological processes that are closely related to the overall nitrate response: "nitrate transport" (GO:0015706), "nitrate assimilation" (GO:0042128), and "response to nitrate" (GO:0010167). Clicking the "View query list" button opens a window that displays the GO-terms and genes that have been added to the query list, which in our case correspond to 231 genes (Figure 4.7C).

In some cases, GO-terms can contain unreliable annotations that can add noise to the predictions. To overcome this, GENIUS allows users to select which GO evidence codes wants to use when importing GO annotations. This can be done by clicking the “Filter GO annotations” button (Figure 4.7A), which opens a window where users can select the evidences to use (Figure 4.7B). By default, GENIUS includes all evidences except “inferred from electronic annotation” (IEA). In this example, we also discard annotations with RCA evidence (inferred from reviewed computational analysis), which add a lot of unreliable annotations for these particular GO-terms. By doing this, we obtain a query list with 38 genes.

Once the query list is defined, clicking “Next” in the main screen (Figure 4.7A) opens a new window that allows the user to introduce a name to identify the prediction and to select the stringency level of the prediction (Figure 4.7D). In most cases, users should use the normal (default) setting. However, users can also run predictions using relaxed or stringent settings to obtain a more exploratory or focused network prediction, respectively. For the purpose of this demonstration, we select the normal setting and start the prediction by clicking “Next”.

Due to the high computational complexity of searching expression signatures among thousands of genes and features, the prediction process can take minutes to several hours, depending on the number of genes in the query list and the size of the expression data available for the selected organism. To alleviate this problem, users can register their email in the start screen, so that GENIUS can send them a notification and a link to the results when the analysis is finished.

Analyzing the results to obtain candidate genes

The results screen provides a network graph view and several tables, organized in tabs, where users can extract relevant information from the predicted network (Figure 4.7).

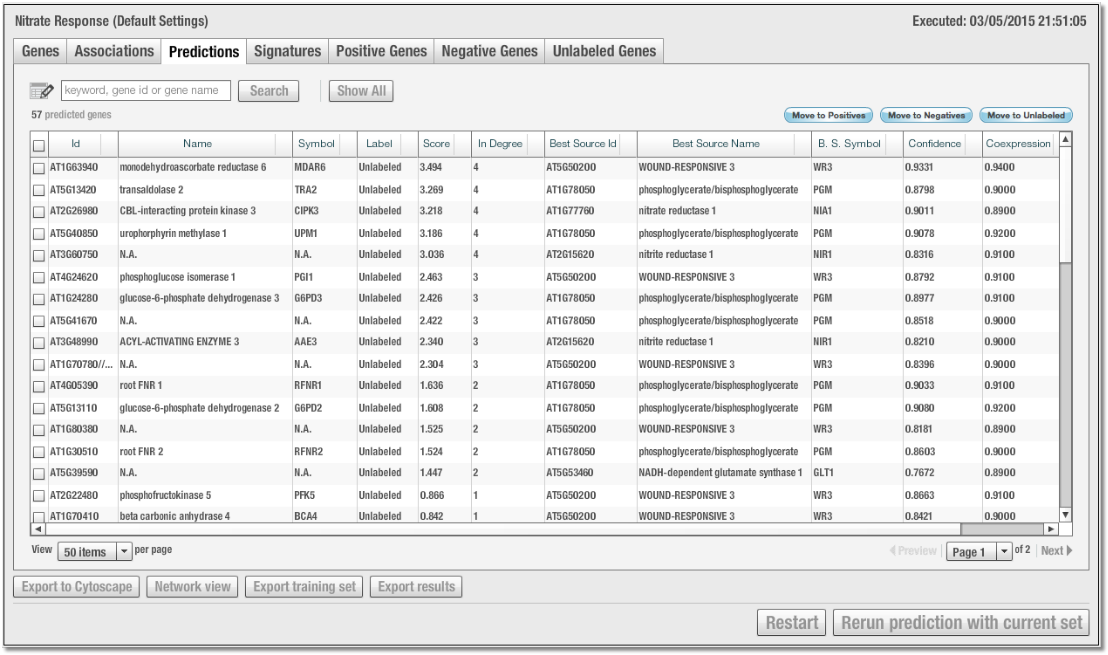
i. Genes tab: provides details about each gene/node of the network, including centrality indicators to rank them according to their relevance.

ii. Associations tab: provides details about each association/edge of the network, including their confidence and coexpression level.

iii. Predictions tab: since each gene can be predicted by various other genes, this tab provides summarized details about each gene predicted in the network, including a score to rank them according to their overall (total) confidence.

iv. Signatures: provides details about the expression signatures used to make the predictions, including the experimental conditions selected and a score to rank them according to how good they can discriminate positive from negative genes.

v. Positive, Negative and Unlabeled Genes tabs: provide details about the positive and negative genes used to train the algorithm, as well as the unlabeled genes that were not used for it.



Nitrate Response (Default Settings)

Executed: 03/05/2015 21:51:05

57 predicted genes

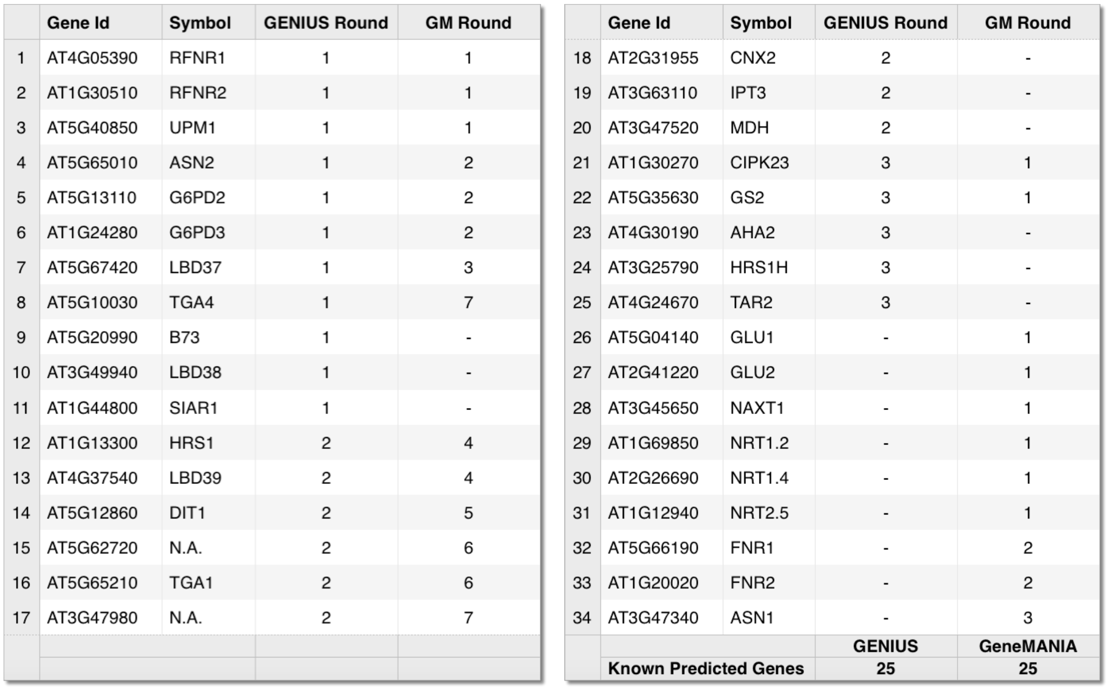
□	Id	Name	Symbol	Label	Score	In Degree	Best Source Id	Best Source Name	B. S. Symbol	Confidence	Coexpression
□	AT1G63940	monodehydroascorbate reductase 6	MDAR6	Unlabeled	3.494	4	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.9331	0.9400
□	AT5G13420	transaldolase 2	TRA2	Unlabeled	3.269	4	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.8798	0.9000
□	AT2G26980	CBL-interacting protein kinase 3	CIPK3	Unlabeled	3.218	4	AT1G77760	nitrate reductase 1	NIA1	0.9011	0.8900
□	AT5G40850	urophorphyrin methylase 1	UPM1	Unlabeled	3.186	4	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.9078	0.9200
□	AT3G60750	N.A.	N.A.	Unlabeled	3.036	4	AT2G15620	nitrite reductase 1	NIR1	0.8316	0.9100
□	AT4G24620	phosphoglucose isomerase 1	PGI1	Unlabeled	2.463	3	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.8792	0.9100
□	AT1G24280	glucose-6-phosphate dehydrogenase 3	G6PD3	Unlabeled	2.426	3	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.8977	0.9100
□	AT5G41670	N.A.	N.A.	Unlabeled	2.422	3	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.8518	0.9000
□	AT3G48990	ACYL-ACTIVATING ENZYME 3	AAE3	Unlabeled	2.340	3	AT2G15620	nitrite reductase 1	NIR1	0.8210	0.9000
□	AT1G70780//...	N.A.	N.A.	Unlabeled	2.304	3	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.8396	0.9000
□	AT4G05390	root FNR 1	RFNR1	Unlabeled	1.636	2	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.9033	0.9100
□	AT5G13110	glucose-6-phosphate dehydrogenase 2	G6PD2	Unlabeled	1.608	2	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.9080	0.9200
□	AT1G80380	N.A.	N.A.	Unlabeled	1.525	2	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.8181	0.8900
□	AT1G30510	root FNR 2	RFNR2	Unlabeled	1.524	2	AT1G78050	phosphoglycerate/bisphosphoglycerate	PGM	0.8603	0.9000
□	AT5G39590	N.A.	N.A.	Unlabeled	1.447	2	AT5G53460	NADH-dependent glutamate synthase 1	GLT1	0.7672	0.8900
□	AT2G22480	phosphofructokinase 5	PFK5	Unlabeled	0.866	1	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.8663	0.9100
□	AT1G70410	beta carbonic anhydrase 4	BCA4	Unlabeled	0.842	1	AT5G50200	WOUND-RESPONSIVE 3	WR3	0.8421	0.9000

Figure 4.8. Results view of GENIUS, showing the details of the predicted genes in the network.

In this case study we focus our attention in the "Predictions" tab, which shows properties like coexpression, confidence and score for each predicted gene, as well as the in-degree of their respective nodes (Figure 4.8). Details about these properties can be seen in Section 3.5.4. By clicking the headers of these properties users can easily rank the predicted genes and prioritize them for functional assays. By default, GENIUS shows the new predictions (unlabeled genes) first, sorted by their prediction score. Unlabeled genes are genes that were not included in the query list, and thus, the predicted ones represent new candidate genes to be involved in BF.

In our case study, the results show a network containing 63 genes. 31 out of the 63 genes are from the positive set (genes in the query list) and 32 are genes from the unlabeled set (new predicted genes). Interestingly, searching the literature we found that 11 of these 32 predicted genes are true positives. We consider as true positives the predicted genes coming from the unlabeled set that have experimental evidence in the literature directly linking them to nitrate responses (Table 4.3, round 1).

Table 4.3. Genes predicted by GENIUS and GeneMANIA (GM) for nitrate response in Arabispsis thaliana that are validated by published literature.



	Gene Id	Symbol	GENIUS Round	GM Round
1	AT4G05390	RFNR1	1	1
2	AT1G30510	RFNR2	1	1
3	AT5G40850	UPM1	1	1
4	AT5G65010	ASN2	1	2
5	AT5G13110	G6PD2	1	2
6	AT1G24280	G6PD3	1	2
7	AT5G67420	LBD37	1	3
8	AT5G10030	TGA4	1	7
9	AT5G20990	B73	1	-
10	AT3G49940	LBD38	1	-
11	AT1G44800	SIAR1	1	-
12	AT1G13300	HRS1	2	4
13	AT4G37540	LBD39	2	4
14	AT5G12860	DIT1	2	5
15	AT5G62720	N.A.	2	6
16	AT5G65210	TGA1	2	6
17	AT3G47980	N.A.	2	7
18	AT2G31955	CNX2	2	-
19	AT3G63110	IPT3	2	-
20	AT3G47520	MDH	2	-
21	AT1G30270	CIPK23	3	1
22	AT5G35630	GS2	3	1
23	AT4G30190	AHA2	3	-
24	AT3G25790	HRS1H	3	-
25	AT4G24670	TAR2	3	-
26	AT5G04140	GLU1	-	1
27	AT2G41220	GLU2	-	1
28	AT3G45650	NAXT1	-	1
29	AT1G69850	NRT1.2	-	1
30	AT2G26690	NRT1.4	-	1
31	AT1G12940	NRT2.5	-	1
32	AT5G66190	FNR1	-	2
33	AT1G20020	FNR2	-	2
34	AT3G47340	ASN1	-	3
			GENIUS	**GeneMANIA**
	Known Predicted Genes		**25**	**25**

In particular, *UPM1*, *G6PD2, G6PD3*, *LBD37*, *LBD38* and *TGA4* stand out as important genes for this process. *UPM1* encodes an enzyme involved in siroheme biosynthesis (Leustek et al., 1997), an essential cofactor for NIR1 (Tripathy, Sherameti, & Oelmuller, 2010). *NIR1*, which appears as one of the top scored *positive* genes of the network, encodes an enzyme that catalyzes the reduction of nitrite to ammonium, a second and critical step in the nitrate reduction pathway (Crawford & Forde, 2002). G6PD2 and G6PD3 are related enzymes that may provide reducing power required for nitrate reduction and ammonia assimilation (Esposito, Massaro, Vona, Di Martino Rigano, & Carfagna, 2003; Wright, Huppe, & Turpin, 1997). In addition, both *NIR1* and *GOGAT* are under the control of LBD37/38 transcription factors (Rubin, Tohge, Matsuda, Saito, & Scheible, 2009). These transcription factors are important regulators of the nitrate response in *Arabidopsis* since they are able to repress the expression of genes related to nitrate transport and assimilation (Rubin et al., 2009). In fact, *LBD37* or *LBD38* overexpressing plants display reduced nitrate content, amino acids and growth (Rubin et al., 2009). Finally, our laboratory recently demonstrated that TGA4, and the closely related TGA1, are important regulatory factors of the root response to nitrate treatments in *Arabidopsis thaliana* (Alvarez et al., 2014).

Using an iterative workflow to refine predictions

In order to facilitate knowledge discovery, GENIUS allows users to incorporate newly predicted genes into the query list and start a new prediction directly from the results view. To do this, users can select the rows of the desired predicted genes in the “Predictions” table and then press the ‘Move to Positives’ button (Figure 4.8). Then, users can start a new prediction by pressing the ‘Rerun prediction with current set’ button (Figure 4.8). This process can be repeated iteratively until there are no new true positive genes.

In our case, we move the 11 true positive genes and then start a new prediction. The inclusion of these new genes allows GENIUS to predict 9 additional true positive genes in the new prediction (Table 4.3, round 2). For example, *LBD39* and *TGA1* are predicted thanks to the added genes *LBD37/38* and *TGA4*, respectively. This means that GENIUS found expression signatures for *LBD37/38* and *TGA4* during its training process, which then predicted *LBD39* and *TGA1*, respectively. The details about the associations predicted between these and other genes can be seen in the

"Associations" tab or as edges in the network view. By applying this process iteratively GENIUS is able to predict a total of 25 true positive genes in 3 rounds of predictions (Table 4.3). The network predicted in the fourth round does not show any additional genes known to participate in the nitrate response. However, it is the one predicted with the most complete query list, and thus, the one that should be used for further analyses.

Comparing GENIUS and GeneMANIA predictions

In order to put GENIUS results in perspective, we compare its results with similar results obtained using GeneMANIA (Zuberi et al., 2013). GeneMANIA does not allow users to directly add GO-terms to its query list. Therefore, we use the genes of the query list generated by GENIUS, which can be exported as an Excel file from the "Query List" window (Figure 4.7A). Surprisingly, using the default settings of GeneMANIA, we are able to predict only 3 true positive genes in the first round of predictions and no additional true positive genes in a second round. In order to try to improve this result, we customized the settings of GeneMANIA by adding all the expression data sets it has available and increasing the number of new genes (predictions) to include in the network from 20 to 50. These settings allow GeneMANIA to find 11 true positive genes in the first round and a total of 25 true positives in 7 rounds (Table 4.3).

Comparing the predictions of both tools reveals that they share 16 true positive genes, while 9 are exclusive to GENIUS and 9 to GeneMANIA (Table 4.3). For GeneMANIA, these 9 genes are related to nitrate transport and metabolism. GENIUS also predicts genes related to nitrogen transport (*AHA2* , *SIAR1* (Ladwig et al., 2012)) and metabolism (*MDH* (Selinski & Scheibe, 2014)), but it additionally includes genes related to transcriptional regulation (*LBD38* (Rubin et al., 2009), *HHO1* (Medici et al., 2015)) and hormone biosynthesis (*TAR2* (Ma et al., 2014), *IPT3* (Takei et al., 2004)). Among GENIUS predictions, HHO1 stands out as a key transcription factor in the early nitrate response (Medici et al., 2015). This transcription factor and its closely related homolog HRS1, control primary root growth depending on the availability of nitrate and phosphate (Medici et al., 2015). Another interesting example is IPT3, an enzyme that catalyzes the initial step in the biosynthesis of cytokinin, a phytohormone that regulates a variety of processes in plant growth and development. Cytokinin concentration in plants is closely related to nitrogen availability (Kamada-Nobusada, Makita, Kojima, & Sakakibara, 2013), *IPT3* expression is specifically regulated by nitrate and a loss-of-function *ipt3* mutation severely diminished the nitrate-dependent accumulation of cytokinin (Takei et al., 2004).

These results show that GENIUS is a powerful tool to predict new genes involved in a biological function of interest and to aid biologists to prioritize genes for functional assays. Our example shows that GENIUS is able to correctly predict the same number of genes than GeneMANIA, but in less iterations and without needing any customization. Also, it exposes one of the main advantages of GENIUS over other tools like GeneMANIA, in which the user needs to select the experimental data and the number of new predicted genes to include in the networks. In contrast, GENIUS automatically selects them based on the genes that the user includes in the query list.

Extracting additional insights from expression signatures

A key aspect of GENIUS is its ability to automatically select genes and experimental conditions containing discriminative expression patterns or “expression signatures”. Examining the discovered signatures can be useful to identify key genes for the biological function of interest, as well as experimental conditions under which these genes may be experimentally verified. This information can be accessed from the “Signatures” tab, which shows a table with the positive genes containing an expression signature, sorted by their expression signature score (*ESS*) (Equation 3.1). This score measures the capability of an expression signature to discriminate positive genes from negative ones, and thus, it can highlight relevant genes for the biological function of interest. In addition, users can see detailed information about the experimental conditions that define each expression signature by clicking on the magnifying glass icon on the left side of each row of the grid.

In our example, the results of the final prediction show that *G6PD3* and *NIA1* have the expression signatures with the highest scores. *NIA1* is a central gene of the nitrate response, as it is involved in the first step of nitrate reduction (Crawford & Forde, 2002). Also, both *G6PD3* and *NIA1* are among the top 15 most consistent nitrate-responsive genes reported in a recent meta-analysis (Canales et al., 2014). In fact, these genes are considered prototypical nitrate responsive genes (Krouk, Mirowski, LeCun, Shasha, & Coruzzi, 2010; Ruffel et al., 2011). Consistent with this observation, examination of the experimental conditions selected for the *G6PD3* expression signature shows that the experiments with highest relevance score correspond to nitrate treated root cells. This shows that GENIUS is able to automatically choose genes and experimental conditions that are relevant for the predicted process, relying solely in the query list of genes the provided by the user as input.

4.4.2. Finding key regulators for complex traits

Biologists often pursue the difficult task of finding key genes to modulate complex traits of interest (e.g. disease, growth, yield, plant nitrogen-use efficiency, water-use efficiency). In Section 4.2, we showed that the former DLS algorithm (Puelma et al., 2012) was able to correctly predict a key gene to modulate nitrogen use efficiency of plants, which is a complex trait involving several biological processes. Here, we illustrate how GENIUS can be successfully used for this purpose by applying it to drought tolerance in *A. thaliana* and comparing its results with the ones of GeneMANIA.

We start the analysis by defining a query list with 11 biological processes that, based in our current knowledge, are involved in drought tolerance. The included GO-terms are: water transport (GO:0006833), response to osmotic stress (GO:0006970), response to water deprivation (GO:0009414), response to water (GO:0009415), plasmodesma organization (GO:0009663), cellular water homeostasis (GO:0009992), guard cell differentiation (GO:0010052), stomatal movement (GO:0010118), response to desiccation (GO:0009269), response to salt stress

(GO:0009651), and abscisic acid transport (GO:0080168). Using these GO-terms, GENIUS generates a query list containing 1,043 genes. Given the large number of query genes, in this case we use the stringent setting to reduce the size of the predicted network and focus it on the most reliable associations.

We also perform a similar analysis using GeneMANIA, using its default settings (it does not allow adjusting stringency), and the list of 1,043 query genes generated by GENIUS as input. Notice that the web version of GeneMANIA does not allow users to make predictions with a query list of this size. Thus, in this case we use the Cytoscape plugin of GeneMANIA, which allows to perform more advanced analyses and supports query lists of any size.

Analyzing the results obtained with both tools, we can see that the network predicted by GENIUS has a hierarchical scale-free network topology, which is not the case for the network predicted by GeneMANIA (Figure 4.9). Gene networks, as most biological networks, usually have a hierarchical and scale-free topology, where most nodes have a small number of connections (degree) and only a few nodes are highly connected (hubs) (Barabási & Oltvai, 2004). GENIUS predicts a network containing 15,763 associations among 2,125 genes (~7.41 associations per gene in average), showing a degree distribution that resembles the ones of scale-free networks (Figure 4.8). Of its genes, 735 are from the positive set (query list) and 1,390 are from the unlabeled set (new predictions). In contrast, GeneMANIA predicts a highly connected network, containing more than 650,000 associations among 1,060 genes (~613 associations per gene), with a degree distribution that greatly differs from scale-free networks. The advantage of obtaining a network with a scale-free topology is that users can take advantage of network theory and use centrality indicators to pinpoint relevant genes. This can be done easily in the "Genes" tab displayed in the results screen, which shows all the genes in the network and several properties that can be used to rank them and highlight relevant genes (See Section 3.5.4 for details of these properties). These properties include three centrality indicators: degree centrality (DC), betweenness centrality (BC), and overall centrality (OC). By default, GENIUS displays genes ranked by their overall centrality (OC), a custom indicator that GENIUS derives by calculating the geometric mean of the degree and betweenness centralities (Section 3.5.4).

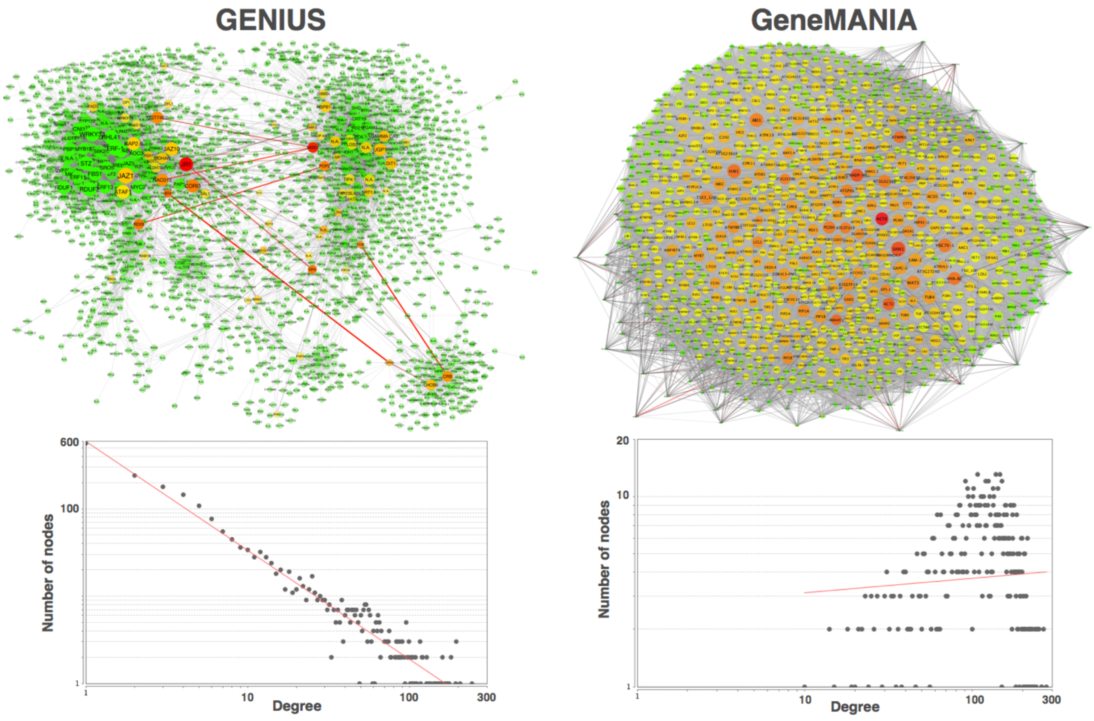
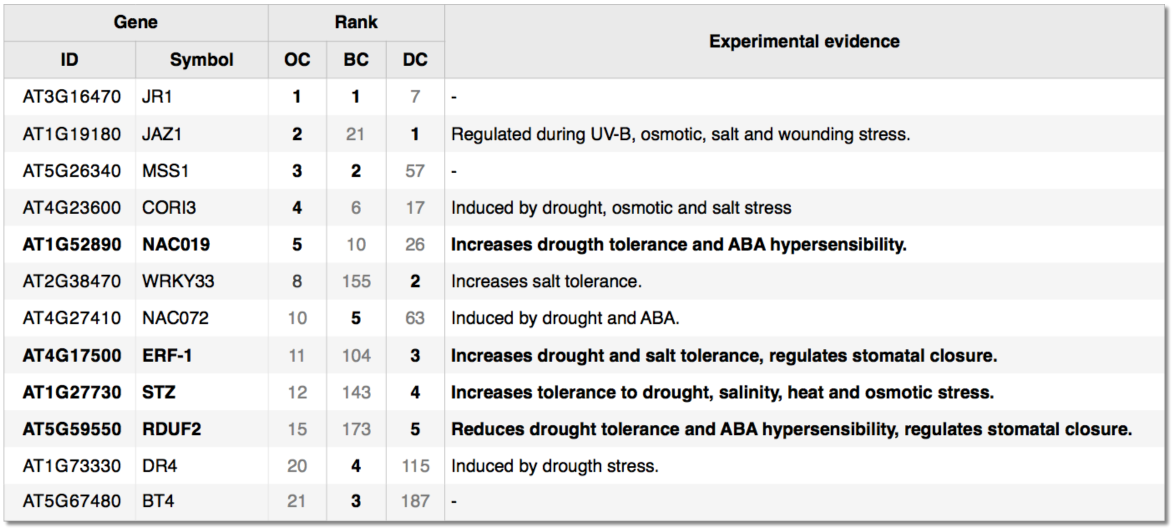


Figure 4.9. Topology of the networks predicted for drought tolerance in A. thaliana.

To show the usefulness of the three centrality indicators, we selected the top 5 ranked genes according to each of them and then searched the literature for experimental evidence linking them to drought tolerance (Table 4.4). By doing this, we obtain a table with 12 genes. Of these, 4 are known to modulate plant drought tolerance: ERF-1 (3rd DC), STZ (4th DC), RDUF2 (5th DC), and NAC019 (5th OC) (M.-C. Cheng, Liao, Kuo, & Lin, 2013; S. J. Kim, Ryu, & Kim, 2012; M et al., 2010; Sakamoto et al., 2004; Schmidt et al., 2013). Available evidence shows that these genes participate in a large number of drought related processes, which coincides with their high centrality in the predicted network. For example, it has been shown that NAC019 overexpressing plants have a significantly increased drought tolerance (Tran et al., 2004). In addition, this gene is induced by drought, high salinity, and abscisic acid, and has been identified as a positive regulator of abscisic acid (ABA) signaling, conferring ABA hypersensitivity when ectopically expressed in plants (M et al., 2010). All these are important processes during drought.

For comparison purposes, we also performed this literature search over the top 12 genes predicted by GeneMANIA, based in the score that it provides. Of these, 2 genes are known to modulate drought tolerance: *NPX1* and *DRIP1* (M. J. Kim, Shin, & Schachtman, 2009; Qin et al., 2008).

Table 4.4. Central genes in the network predicted by GENIUS for drought tolerance in Arabidopsis thaliana.



Gene		Rank			Experimental evidence
ID	Symbol	OC	BC	DC	
AT3G16470	JR1	**1**	**1**	7	-
AT1G19180	JAZ1	**2**	21	**1**	Regulated during UV-B, osmotic, salt and wounding stress.
AT5G26340	MSS1	**3**	**2**	57	-
AT4G23600	CORI3	**4**	6	17	Induced by drought, osmotic and salt stress
AT1G52890	**NAC019**	**5**	10	26	**Increases drougth tolerance and ABA hypersensibility.**
AT2G38470	WRKY33	8	155	**2**	Increases salt tolerance.
AT4G27410	NAC072	10	**5**	63	Induced by drought and ABA.
AT4G17500	**ERF-1**	11	104	**3**	**Increases drought and salt tolerance, regulates stomatal closure.**
AT1G27730	**STZ**	12	143	**4**	**Increases tolerance to drought, salinity, heat and osmotic stress.**
AT5G59550	**RDUF2**	15	173	**5**	**Reduces drought tolerance and ABA hypersensibility, regulates stomatal closure.**
AT1G73330	DR4	20	**4**	115	Induced by drougth stress.
AT5G67480	BT4	21	**3**	187	-

There are other genes in the rankings of both tools that, although have not been demonstrated to affect drought tolerance directly, are known to be regulated by different stress conditions that are present during drought, which makes them interesting candidates. In the case of GeneMANIA, we found 4 genes meeting this criterion (TUA2, AT1G61220, PLDP1, and EDL3) (Bargmann et al., 2009; E Stecker, Minkoff, & Sussman, 2014; Jha, Shirley, Tester, & Roy, 2010; Koops et al., 2011; J.-H. Lee, Terzaghi, & Deng, 2011; Seo et al., 2014), while in the case of GENIUS we found 5 genes (JAZ1, CORI3, WRKY33, NAC072, and DR4) (Gosti, Bertauche, Vartanian, & Giraudat, 1995; Hahn et al., 2013; Jiang & Deyholos, 2009; Sadhukhan et al., 2014; Tran et al., 2004). Among these five genes, there is another gene of the NAC transcription factors family, NAC072 (5th BC), which is induced in response to drought, high salinity, ABA, and JA treatments (Tran et al., 2004). More importantly, it has been shown that NAC072-overexpressing transgenic plants are hypersensitive to ABA and that ABA- and stress-inducible genes are up regulated, while the opposite effect is displayed in NAC072-overexpressing transgenic plants (Tran et al., 2004). This evidence suggests that NAC072 is an attractive candidate to improve drought tolerance.

In addition to the genes discussed above, the ranking of GENIUS includes 3 genes for which we could not find any evidence linking them to drought tolerance (*JR1*, *MSS1*, and *BT4*). Nevertheless, their high centrality in the network makes them interesting candidates for future experimental validation.

In order to extract additional insights about the trait or biological function of interest, GENIUS allows users to export its inferred networks to Cytoscape. As an example, in this case study we use the ClusterMaker (Morris et al., 2011) and BiNGO (Maere, Heymans, & Kuiper, 2005) Cytoscape plugins, which allow us to find clusters in the network and GO biological process enriched in each of them, respectively. As we show below, we find biologically meaningful clusters of genes thanks to the hierarchical and scale-free topology of the predicted network. These clusters can help scientist to better understand the biological context in which central genes were predicted as well as the biological processes involved in the trait or biological function of interest.

Our analysis reveals five main clusters in the network (Figure 4.10), enriched in several biological processes that are relevant for drought tolerance, as detailed below. The light blue and yellow clusters are enriched in response to carbohydrate stimulus (GO:0099743) and photosynthesis (GO:0015979) genes, respectively. The green cluster is enriched in response to abscisic acid stimulus (GO:0009737) and response to water deprivation (GO: 0009414), and the blue cluster is enriched in response to jasmonic acid (GO:0009753), sulfur metabolic process (GO:0006790), and glucosinolate metabolic process (GO:0019760). All these processes are known to be relevant during drought tolerance. Water deficits perceived by roots induce *de novo* ABA biosynthesis (Sauter, 2001), which in turn triggers stomatal closure (Boursiac et al., 2013). In a drought scenario, the plant closes the stomata to decrease their evapotranspiration, which, as a consequence, decreases its photosynthetic and carbon reduction cycle activity (M. Ashraf & Harris, 2013; Reddy, Chaitanya, & Vivekanandan, 2004). Jasmonic acid (JA) has a protecting role under hydric stress condition (de Ollas, Hernando, Arbona, & Gómez-Cadenas, 2013) and its early accumulation is necessary for the subsequent ABA increase in roots (H. Du, Liu, & Xiong, 2013). Sulfur has also been associated to the regulation of ABA and an important role under abiotic stresses in general (Wilkinson & Davies, 2002). Sulfur participates in glutathione synthesis, which maintains the cellular redox balance and mitigates damage caused by reactive oxygen species (Gallardo, Courty, Le Signor, Wipf, & Vernoud, 2014). Also, it has been shown that drought conditions induce the synthesis and accumulation of different glucosinolates as part of the plant responses to stress, through a process called osmotic adjustment (Del Carmen Martínez-Ballesta, Moreno, & Carvajal, 2013). During this process, plants accumulate a variety of organic and inorganic substances (sugars, polyols, amino acids, alkaloids and inorganic ions) to reduce osmotic potential and improve cell water retention (Rhodes & Samaras, 1994). Indeed, keeping the cell homeostasis and coping the osmolite changes concomitant with water deficiency is one of the main challenges for plants during drought. In fact, this process is represented in the pink cluster (Figure 4.10), which is enriched in genes from response to osmotic stress (GO:0006970), and response to salt stress (GO:0009651). In addition, salt stress and drought conditions are intricately related, because the decrease in water potential in both abiotic stresses results in a reduced growth of the cell, root, and shoot. Also, reduced water potential causes inhibition of cell expansion and reduction in cell wall synthesis. Both stresses affect metabolism of the cell, such as carbon-reduction cycle, light reactions, energy charge, and proton pumping, as well as the production of toxic molecules. A salinity environment reduces the ability of plants to take up water, and this quickly causes reductions in growth rate, along with a range of metabolic changes comparable to those caused by water stress (Agarwal, Shukla, Gupta, & Jha, 2013; Muhammad Ashraf & Akram, 2009; Flowers, 2004; Hasegawa, Bressan, Zhu, & Bohnert, 2000; Munns, 2005).

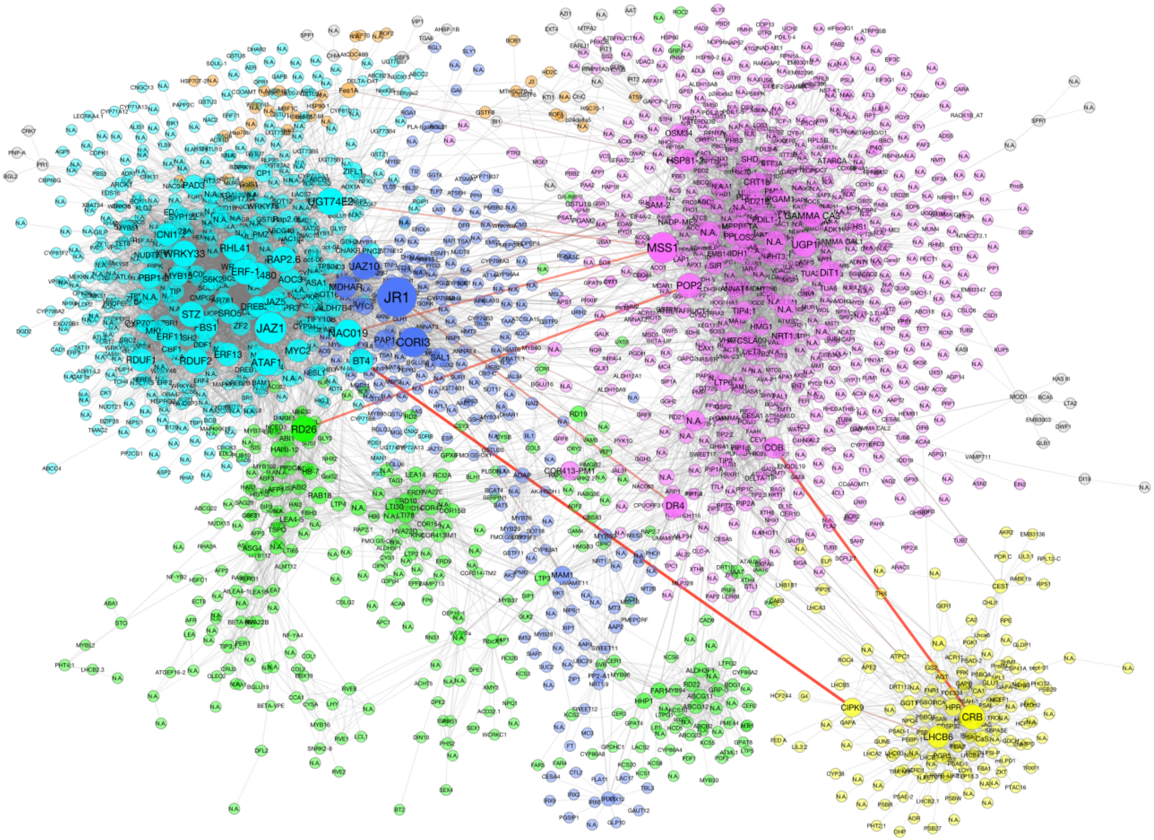


Figure 4.10. Clusters in the network inferred by GENIUS for drought tolerance in Arabidopsis thaliana.

Remarkably, several of the biological processes mentioned above were not directly added to the query list. Nevertheless, the inferred network exposes them as important processes for drought tolerance by organizing them in clusters, which shows the kind of insights (or hypotheses) that can be derived from analyzing the networks predicted by GENIUS.

GENIUS allows users to export its inferred networks to Cytoscape for more advanced analyses. In this case we used ClusterMaker and BINGO Cytoscape plugins to find clusters, displayed in different colors, and GO biological processes enriched in them.

This analysis revealed five main clusters, enriched in various biological processes related to drought tolerance: light blue – "response to carbohydrate stimulus" (GO:0099743); yellow – "photosynthesis" (GO:0015979); green – "response to abscisic acid stimulus" (GO:0009737) and "response to water deprivation" (GO: 0009414); pink – "response to osmotic stress" (GO:0006970) and "response to salt stress" (GO:0009651); blue – "response to jasmonic acid stimulus" (GO:0009753). Interestingly, some of these processes were not included in the gene list used for training but were predicted by GENIUS.

Therefore, this case study shows that GENIUS can successfully aid biologists to discover relevant genes for complex traits of interest, which may involve many biological processes and thousands of genes. It also shows the usefulness of the three centrality indicators provided by GENIUS to pinpoint relevant genes in the inferred network: degree, betweenness, and overall centralities. Finally, they show how GENIUS output can seamlessly integrate with Cytoscape to perform more advanced analysis over the predicted networks, which may reveal additional insights about the trait and its underlying biological processes.

5. CONCLUSIONS

In this thesis, we presented DLS, a novel method that combines supervised machine learning and coexpression approaches to effectively predict gene networks and new genes for a biological function of interest. In addition, we presented GENIUS, a web server and user-friendly tool to allow biologist to use DLS on their own researches.

We developed four key concepts that allow DLS (and GENIUS) to effectively predict gene function:

i. the discovery of false negatives to derive informative training sets,
ii. the supervised search of discriminative expression patterns in subsets of genes and experimental conditions (expression signatures),
iii. a Bayesian probabilistic approach to derive the confidence for each prediction, and
iv. the construction of a discriminative coexpression network to represent predictions.

Our systematic evaluations show that DLS is able to provide gene functional predictions with accuracies comparable to the highly discriminative SVMs, while maintaining the expressiveness of coexpression networks. They also show the effectiveness of our automatic procedure to choose control-test pairs of conditions and derive large expression datasets. Remarkably, they show that, unlike SVMs and coexpression networks, DLS systematically improves its prediction performance as more experimental conditions are added to the dataset generated by our procedure. In addition, they indicate that supervised approaches can predict gene function more effectively than semi-supervised approaches, if informative negative sets can be derived. As a consequence, they emphasize the importance and usefulness of our approach to discover false negatives and refine the training set. Finally, they show that DLS can make high quality predictions in most GO-terms for the eight organisms supported by GENIUS.

Our evaluations in real research scenarios show that GENIUS can be effectively used to make novel discoveries. More specifically, from these evaluations we can extract the following four conclusions.

i. GENIUS can be successfully used to study biological functions ranging from specific biological processes to complex traits of interest, which may involve several biological processes and thousands of genes.
ii. GENIUS can be successfully used to predict new genes for a biological function of interest, including genes that other state-of-the art tool are not able to find.
iii. Expression signatures contain experimental conditions that are biologically coherent and relevant for the biological function of interest, which can further guide biologists to understand the biological context of the predictions and design sound functional assays.
iv. GENIUS can be used to successfully pinpoint key genes to modulate biological functions and complex traits. Remarkably, it pinpointed a novel gene to improve nitrogen use efficiency of plants, which was later validated experimentally. For this, an important aspect is its use of graph theory statistics over the predicted network, in order to rank and prioritize genes for experimental validation.

Hence, we believe the presented method, DLS, and its web interface, GENIUS, can help biologists to generate and explore new hypothesis, make novel discoveries, and ultimately, improve our molecular understanding of biological systems.

BIBLIOGRAPHY

Agarwal, P. K., Shukla, P. S., Gupta, K., & Jha, B. (2013). Bioengineering for salinity tolerance in plants: state of the art. *Molecular Biotechnology*, *54*(1), 102–23. http://doi.org/10.1007/s12033-012-9538-3

Alon, U. (2003). Biological Networks: The Tinkerer as an Engineer. *Science*, *301*(5641), 1866–1867. http://doi.org/10.1126/science.1089072

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America*, *96*(12), 6745–6750. http://doi.org/10.1073/pnas.96.12.6745

Alvarez, J. M., Riveras, E., Vidal, E. A., Gras, D. E., Contreras-Lopez, O., Tamayo, K. P., … Gutierrez, R. A. (2014). Systems approach identifies TGA1 and TGA4 transcription factors as important regulatory components of the nitrate response of Arabidopsis thaliana roots. *Plant J*, *80*(1), 1–13. http://doi.org/10.1111/tpj.12618

Araus, V., Vidal, E. A., Puelma, T., Alamos, S., Mieulet, D., Guiderdoni, E., & Gutiérrez, R. A. (2016). Members of BTB gene family regulate negatively nitrate uptake and nitrogen use efficiency in Arabidopsis thaliana and Oryza sativa. *Plant Physiology*, pp.01731.2015. http://doi.org/10.1104/pp.15.01731

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., … Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics*, *25*(1), 25–29. http://doi.org/10.1038/75556

Ashraf, M., & Akram, N. A. (2009). Improving salinity tolerance of plants through conventional breeding and genetic engineering: An analytical comparison. *Biotechnology Advances*, *27*(6), 744–52. http://doi.org/10.1016/j.biotechadv.2009.05.026

Ashraf, M., & Harris, P. J. C. (2013). Photosynthesis under stressful environments: An overview. *Photosynthetica*, *51*(2), 163–190. http://doi.org/10.1007/s11099-013-0021-

6

Azuaje, F. (2014). Selecting biologically informative genes in co-expression networks with a centrality score. Retrieved from http://www.biomedcentral.com/content/pdf/1745-6150-9-12.pdf

Barabási, A.-L., & Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature Reviews. Genetics*, *5*(2), 101–113. http://doi.org/10.1038/nrg1272

Barakat, N., & Bradley, A. P. (2010). Rule extraction from support vector machines: A review. *Neurocomputing*, *74*(1-3), 178–190. http://doi.org/10.1016/j.neucom.2010.02.016

Bargmann, B. O. R., Laxalt, A. M., ter Riet, B., van Schooten, B., Merquiol, E., Testerink, C., … Munnik, T. (2009). Multiple PLDs required for high salinity and water deficit tolerance in plants. *Plant & Cell Physiology*, *50*(1), 78–89. http://doi.org/10.1093/pcp/pcn173

Barutcuoglu, Z., Schapire, R. E., & Troyanskaya, O. G. (2006). Hierarchical multi-label prediction of gene function. *Bioinformatics*, *22*(7), 830–836. http://doi.org/10.1093/bioinformatics/btk048

Bassel, G. W., Lan, H., Glaab, E., Gibbs, D. J., Gerjets, T., Krasnogor, N., … Provart, N. J. (2011). Genome-wide network model capturing seed germination reveals coordinated regulation of plant cellular phase transitions. *Proceedings of the National Academy of Sciences*, *108*(23), 9709–9714. http://doi.org/10.1073/pnas.1100958108

Ben-Dor, A., Chor, B., Karp, R., & Yakhini, Z. (2003). Discovering local structure in gene expression data: the order-preserving submatrix problem. *Journal of Computational Biology : A Journal of Computational Molecular Cell Biology*, *10*(3-4), 373–384. http://doi.org/10.1089/10665270360688075

Bolstad, B. M., Irizarry, R. A., Astrand, M., Speed, T. P., & Astrand, M. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics (Oxford, England)*, *19*(2), 185–193.

http://doi.org/10.1093/bioinformatics/19.2.185

Boursiac, Y., Léran, S., Corratgé-Faillie, C., Gojon, A., Krouk, G., & Lacombe, B. (2013). ABA transport and transporters. *Trends in Plant Science*, *18*(6), 325–33. http://doi.org/10.1016/j.tplants.2013.01.007

Breitling, R., Armengaud, P., Amtmann, A., & Herzyk, P. (2004). Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Letters*, *573*(1-3), 83–92. http://doi.org/10.1016/j.febslet.2004.07.055

Brown, M. M. P. S., Grundy, W. W. N., Lin, D., Cristianini, N., Sugnet, C. W. C., Furey, T. T. S., … Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences of the United States of America*, *97*(1), 262–267. http://doi.org/10.1073/pnas.97.1.262

Canales, J., Moyano, T. C., Villarroel, E., & Gutierrez, R. A. (2014). Systems analysis of transcriptome data provides new hypotheses about Arabidopsis root response to nitrate treatments. *Front Plant Sci*, *5*, 22. http://doi.org/10.3389/fpls.2014.00022

Chang, C. C., & Lin, C. J. (2001). LIBSVM: a library for support vector machines. Retrieved from http://www.csie.ntu.edu.tw/~cjlin/libsvm

Cheng, M.-C., Liao, P.-M., Kuo, W.-W., & Lin, T.-P. (2013). The Arabidopsis ETHYLENE RESPONSE FACTOR1 regulates abiotic stress-responsive gene expression by binding to different cis-acting elements in response to different stress signals. *Plant Physiology*, *162*(3), 1566–82. http://doi.org/10.1104/pp.113.221911

Cheng, Y., & Church, G. M. (2000). Biclustering of expression data. *Proceedings / ... International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology*, *8*, 93–103. Retrieved from http://view.ncbi.nlm.nih.gov/pubmed/10977070

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. http://doi.org/10.1007/BF00994018

Crawford, N. M., & Forde, B. G. (2002). Molecular and developmental biology of inorganic nitrogen nutrition. *Arabidopsis Book*, *1*, e0011. http://doi.org/10.1199/tab.0011

Dawson, C. J., & Hilton, J. (2011). Fertiliser availability in a resource-limited world: Production and recycling of nitrogen and phosphorus. *Food Policy*, *36*, S14–S22. http://doi.org/10.1016/j.foodpol.2010.11.012

de Ollas, C., Hernando, B., Arbona, V., & Gómez-Cadenas, A. (2013). Jasmonic acid transient accumulation is needed for abscisic acid increase in citrus roots under drought stress conditions. *Physiologia Plantarum*, *147*(3), 296–306. http://doi.org/10.1111/j.1399-3054.2012.01659.x

Del Carmen Martínez-Ballesta, M., Moreno, D. A., & Carvajal, M. (2013). The physiological importance of glucosinolates on plant response to abiotic stress in Brassica. *International Journal of Molecular Sciences*, *14*(6), 11607–25. http://doi.org/10.3390/ijms140611607

Du, H., Liu, H., & Xiong, L. (2013). Endogenous auxin and jasmonic acid levels are differentially modulated by abiotic stresses in rice. *Frontiers in Plant Science*, *4*, 397. http://doi.org/10.3389/fpls.2013.00397

Du, L., & Poovaiah, B. W. (2004). A novel family of Ca 2+/calmodulin-binding proteins involved in transcriptional regulation: Interaction with fsh/Ring3 class transcription activators. *Plant Molecular Biology*, *54*, 549–569. http://doi.org/10.1023/B:PLAN.0000038269.98972.bb

E Stecker, K., Minkoff, B. B., & Sussman, M. R. (2014). Phosphoproteomic Analyses Reveal Early Signaling Events in the Osmotic Stress Response. *Plant Physiology*, *165*(3), 1171–1187. http://doi.org/10.1104/pp.114.238816

Edgar, R., Domrachev, M., & Lash, A. E. (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, *30*(1), 207–210. http://doi.org/10.1093/nar/30.1.207

Eisen, M. B., Spellman, P. T., Brown, P. O., & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of*

Sciences, *95*(25), 14863–14868. http://doi.org/10.1073/pnas.95.25.14863

Esposito, S., Massaro, G., Vona, V., Di Martino Rigano, V., & Carfagna, S. (2003). Glutamate synthesis in barley roots: the role of the plastidic glucose-6-phosphate dehydrogenase. *Planta*, *216*(4), 639–647. http://doi.org/10.1007/s00425-002-0892-4

Flowers, T. J. (2004). Improving crop salt tolerance. *Journal of Experimental Botany*, *55*(396), 307–19. http://doi.org/10.1093/jxb/erh003

Franceschini, A., Szklarczyk, D., Frankild, S., Kuhn, M., Simonovic, M., Roth, A., … Jensen, L. J. (2013). STRING v9.1: protein-protein interaction networks, with increased coverage and integration. *Nucleic Acids Research*, *41*, D808–15. http://doi.org/10.1093/nar/gks1094

Fung, G., Sandilya, S., & Rao, R. B. (2005). Rule extraction from linear support vector machines. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (pp. 32–40). Chicago, Illinois, USA: ACM. http://doi.org/10.1145/1081870.1081878

Gallardo, K., Courty, P.-E., Le Signor, C., Wipf, D., & Vernoud, V. (2014). Sulfate transporters in the plant's response to drought and salinity: regulation and possible functions. *Frontiers in Plant Science*, *5*, 580. http://doi.org/10.3389/fpls.2014.00580

Gosti, F., Bertauche, N., Vartanian, N., & Giraudat, J. (1995). Abscisic acid-dependent and -independent regulation of gene expression by progressive drought in Arabidopsis thaliana. *Mgg Molecular & General Genetics*, *246*(1), 10–18. http://doi.org/10.1007/BF00290128

Gutierrez, R. A. (2012). Systems biology for enhanced plant nitrogen nutrition. *Science*, *336*(6089), 1673–1675. http://doi.org/10.1126/science.1217620

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene Selection for Cancer Classification using Support Vector Machines. *Mach. Learn.*, *46*(1-3), 389–422. http://doi.org/10.1023/A:1012487302797

Hahn, A., Kilian, J., Mohrholz, A., Ladwig, F., Peschke, F., Dautel, R., … Wanke, D. (2013). Plant Core Environmental Stress Response Genes Are Systemically Coordinated during

Abiotic Stresses. *International Journal of Molecular Sciences*, *14*(4), 7617–41. http://doi.org/10.3390/ijms14047617

Hasegawa, P. M., Bressan, R. A., Zhu, J.-K., & Bohnert, H. J. (2000). PLANT CELLULAR AND MOLECULAR RESPONSES TO HIGH SALINITY. *Annual Review of Plant Physiology and Plant Molecular Biology*, *51*, 463–499. http://doi.org/10.1146/annurev.arplant.51.1.463

Horan, K., Jang, C., Bailey-Serres, J., Mittler, R., Shelton, C., Harper, J. F., … Girke, T. (2008). Annotating Genes of Known and Unknown Function by Large-Scale Coexpression Analysis. *Plant Physiol.*, *147*(1), 41–57. http://doi.org/10.1104/pp.108.117366

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics (Oxford, England)*, *4*(2), 249–264. http://doi.org/10.1093/biostatistics/4.2.249

Jansen, R., & Gerstein, M. (2004). Analyzing protein function on a genomic scale: the importance of gold-standard positives and negatives for network prediction. *Current Opinion in Microbiology*, *7*(5), 535–545. http://doi.org/10.1016/j.mib.2004.08.012

Jha, D., Shirley, N., Tester, M., & Roy, S. J. (2010). Variation in salinity tolerance and shoot sodium accumulation in Arabidopsis ecotypes linked to differences in the natural expression levels of transporters involved in sodium transport. *Plant, Cell & Environment*, *33*(5), 793–804. http://doi.org/10.1111/j.1365-3040.2009.02105.x

Jiang, Y., & Deyholos, M. K. (2009). Functional characterization of Arabidopsis NaCl-inducible WRKY25 and WRKY33 transcription factors in abiotic stresses. *Plant Molecular Biology*, *69*(1-2), 91–105. http://doi.org/10.1007/s11103-008-9408-3

Ju, X.-T., Xing, G.-X., Chen, X.-P., Zhang, S.-L., Zhang, L.-J., Liu, X.-J., … Zhang, F.-S. (2009). Reducing environmental risk by improving N management in intensive Chinese agricultural systems. *Proceedings of the National Academy of Sciences of the United States of America*, *106*(9), 3041–6. http://doi.org/10.1073/pnas.0813417106

Jupiter, D., Chen, H., & VanBuren, V. (2009). STARNET 2: a web-based tool for accelerating discovery of gene regulatory networks using microarray co-expression data. *BMC Bioinformatics*, *10*(1), 332. http://doi.org/10.1186/1471-2105-10-332

Kamada-Nobusada, T., Makita, N., Kojima, M., & Sakakibara, H. (2013). Nitrogen-dependent regulation of de novo cytokinin biosynthesis in rice: the role of glutamine metabolism as an additional signal. *Plant Cell Physiol*, *54*(11), 1881–1893. http://doi.org/10.1093/pcp/pct127

Kao, H.-L., & Gunsalus, K. C. (2008). Browsing multidimensional molecular networks with the generic network browser (N-Browse). *Current Protocols in Bioinformatics / Editoral Board, Andreas D. Baxevanis ... [et Al.]*, *Chapter 9*, Unit 9.11. http://doi.org/10.1002/0471250953.bi0911s23

Katari, M. S., Nowicki, S. D., Aceituno, F. F., Nero, D., Kelfer, J., Thompson, L. P., … Gutiérrez, R. a. (2010). VirtualPlant: a software platform to support systems biology research. *Plant Physiology*, *152*(2), 500–15. http://doi.org/10.1104/pp.109.147025

Kim, M. J., Shin, R., & Schachtman, D. P. (2009). A Nuclear Factor Regulates Abscisic Acid Responses in Arabidopsis. *PLANT PHYSIOLOGY*. http://doi.org/10.1104/pp.109.144766

Kim, S. J., Ryu, M. Y., & Kim, W. T. (2012). Suppression of Arabidopsis RING-DUF1117 E3 ubiquitin ligases, AtRDUF1 and AtRDUF2, reduces tolerance to ABA-mediated drought stress. *Biochemical and Biophysical Research Communications*, *420*(1), 141–7. http://doi.org/10.1016/j.bbrc.2012.02.131

Kim, S. K., Lund, J., Kiraly, M., Duke, K., Jiang, M., Stuart, J. M., … Davidson, G. S. (2001). A Gene Expression Map for Caenorhabditis elegans. *Science*, *293*(5537), 2087–2092. http://doi.org/10.1126/science.1061603

Koops, P., Pelser, S., Ignatz, M., Klose, C., Marrocco-Selden, K., & Kretsch, T. (2011). EDL3 is an F-box protein involved in the regulation of abscisic acid signalling in Arabidopsis thaliana. *Journal of Experimental Botany*, *62*(15), 5547–60. http://doi.org/10.1093/jxb/err236

Krouk, G., Mirowski, P., LeCun, Y., Shasha, D. E., & Coruzzi, G. M. (2010). Predictive network modeling of the high-resolution dynamic plant transcriptome in response to nitrate. *Genome Biol*, *11*(12), R123. http://doi.org/10.1186/gb-2010-11-12-r123

Kuramochi, M., & Karypis, G. (2001). Gene Classification using Expression Profiles: A Feasibility Study, 1–20. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.20.8514

Ladwig, F., Stahl, M., Ludewig, U., Hirner, A. A., Hammes, U. Z., Stadler, R., … Koch, W. (2012). Siliques Are Red1 from Arabidopsis Acts as a Bidirectional Amino Acid Transporter That Is Crucial for the Amino Acid Homeostasis of Siliques. *PLANT PHYSIOLOGY*. http://doi.org/10.1104/pp.111.192583

Lanckriet, G. R. G., Deng, M., Cristianini, N., Jordan, M. I., & Noble, W. S. (2004). Kernel-based data fusion and its application to protein function prediction in yeast. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, *311*, 300–311. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/14992512

Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., … Robles, V. (2006). Machine learning in bioinformatics. *Briefings in Bioinformatics*, *7*(1), 86–112. http://doi.org/10.1093/bib/bbk007

Lassaletta, L., Billen, G., Grizzetti, B., Garnier, J., Leach, A. M., & Galloway, J. N. (2014). Food and feed trade as a driver in the global nitrogen cycle: 50-year trends. *Biogeochemistry*, *118*(1-3), 225–241. http://doi.org/10.1007/s10533-013-9923-4

Lee, I., Ambaru, B., Thakkar, P., Marcotte, E. M., & Rhee, S. Y. (2010). Rational association of genes with traits using a genome-scale gene network for Arabidopsis thaliana. *Nature Biotechnology*, *28*(2), 149–156. http://doi.org/10.1038/nbt.1603

Lee, J.-H., Terzaghi, W., & Deng, X. W. (2011). DWA3, an Arabidopsis DWD protein, acts as a negative regulator in ABA signal transduction. *Plant Science : An International Journal of Experimental Plant Biology*, *180*(2), 352–7. http://doi.org/10.1016/j.plantsci.2010.10.008

Leustek, T., Smith, M., Murillo, M., Singh, D. P., Smith, A. G., Woodcock, S. C., … Warren,

M. J. (1997). Siroheme biosynthesis in higher plants. Analysis of an S-adenosyl-L-methionine-dependent uroporphyrinogen III methyltransferase from Arabidopsis thaliana. *J Biol Chem*, *272*(5), 2744–2752. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/9006913

M, J., T, K., M, N., P, G., K, P., C, O., & K, S. (2010). The Arabidopsis thaliana NAC transcription factor family: structure-function relationships and determinants of ANAC019 stress signalling. Retrieved from http://www.biochemj.org/bj/426/bj4260183.htm

Ma, W., Li, J., Qu, B., He, X., Zhao, X., Li, B., … Tong, Y. (2014). Auxin biosynthetic gene TAR2 is involved in low nitrogen-mediated reprogramming of root architecture in Arabidopsis. *Plant Journal*, *78*(1), 70–79. http://doi.org/10.1111/tpj.12448

Madeira, S. C. S. C., & Oliveira, A. L. (2004). Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *1*(1), 24–45. http://doi.org/10.1109/TCBB.2004.2

Maere, S., Heymans, K., & Kuiper, M. (2005). BiNGO: a Cytoscape plugin to assess overrepresentation of gene ontology categories in biological networks. *Bioinformatics (Oxford, England)*, *21*(16), 3448–9. http://doi.org/10.1093/bioinformatics/bti551

Mateos, A., Dopazo, J. J. J., Jansen, R., Tu, Y., Gerstein, M., & Stolovitzky, G. (2002). Systematic Learning of Gene Functional Classes From DNA Array Expression Data by Using Multilayer Perceptrons. *Genome Research*, *12*(11), 1703–1715. http://doi.org/10.1101/gr.192502

Medici, A., Marshall-Colon, A., Ronzier, E., Szponarski, W., Wang, R., Gojon, A., … Krouk, G. (2015). AtNIGT1/HRS1 integrates nitrate and phosphate signals at the Arabidopsis root tip. *Nat Commun*, *6*, 6274. http://doi.org/10.1038/ncomms7274

Mitchell, T. M. (1997). *Machine Learning* (1st ed.). McGraw-Hill Science/Engineering/Math. Retrieved from http://www.worldcat.org/isbn/0070428077

Moreau, Y., & Tranchevent, L.-C. (2012). Computational tools for prioritizing candidate genes: boosting disease gene discovery. *Nature Reviews. Genetics*, *13*(8), 523–36.

http://doi.org/10.1038/nrg3253

Morris, J. H., Apeltsin, L., Newman, A. M., Baumbach, J., Wittkop, T., Su, G., … Ferrin, T. E. (2011). clusterMaker: a multi-algorithm clustering plugin for Cytoscape. *BMC Bioinformatics*, *12*, 436. http://doi.org/10.1186/1471-2105-12-436

Mostafavi, S., Ray, D., Warde-Farley, D., Farley, D. W., Grouios, C., & Morris, Q. (2008). GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome …*, *9*(Suppl 1), S4+. http://doi.org/10.1186/gb-2008-9-s1-s4

Munns, R. (2005). Genes and salt tolerance: bringing them together. *The New Phytologist*, *167*(3), 645–63. http://doi.org/10.1111/j.1469-8137.2005.01487.x

Obayashi, T., Nishida, K., Kasahara, K., & Kinoshita, K. (2011). ATTED-II updates: condition-specific gene coexpression to extend coexpression analyses and applications to a broad range of flowering plants. *Plant & Cell Physiology*, *52*(2), 213–219.

Obayashi, T., Okamura, Y., Ito, S., Tadaka, S., Motoike, I. N., & Kinoshita, K. (2013). COXPRESdb: a database of comparative gene coexpression networks of eleven species for mammals. *Nucleic Acids Research*, *41*(Database issue), D1014–20. http://doi.org/10.1093/nar/gks1014

Parzen, E. (1962). On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, *33*(3), 1065–1076. http://doi.org/10.1214/aoms/1177704472

Prelić, A., Bleuler, S., Zimmermann, P., Wille, A., Bühlmann, P., Gruissem, W., … Zitzler, E. (2006). A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, *22*(9), 1122–1129. http://doi.org/10.1093/bioinformatics/btl060

Puelma, T., Gutierrez, R. A., & Soto, A. (2012). Discriminative local subspaces in gene expression data for effective gene function prediction. *Bioinformatics*, *28*(17), 2256–2264. http://doi.org/10.1093/bioinformatics/bts455

Qin, F., Sakuma, Y., Tran, L.-S. P., Maruyama, K., Kidokoro, S., Fujita, Y., … Yamaguchi-

Shinozaki, K. (2008). Arabidopsis DREB2A-interacting proteins function as RING E3 ligases and negatively regulate plant drought stress-responsive gene expression. *The Plant Cell*, *20*(6), 1693–1707. http://doi.org/10.1105/tpc.107.057380

Reddy, A. R., Chaitanya, K. V., & Vivekanandan, M. (2004). Drought-induced responses of photosynthesis and antioxidant metabolism in higher plants. *Journal of Plant Physiology*, *161*(11), 1189–1202. http://doi.org/10.1016/j.jplph.2004.01.013

Rhodes, D., & Samaras, Y. (1994). Genetic control of osmoregulation in plants. *Cellular and Molecular Physiology of Cell Volume Regulation*, 347–361.

Robert, H. S., Quint, A., Brand, D., Vivian-Smith, A., & Offringa, R. (2009). BTB and TAZ domain scaffold proteins perform a crucial function in Arabidopsis development. *Plant Journal*, *58*(1), 109–121. http://doi.org/10.1111/j.1365-313X.2008.03764.x

Robertson, G. P., & Vitousek, P. M. (2009). Nitrogen in Agriculture: Balancing the Cost of an Essential Resource. *Annual Review of Environment and Resources*, *34*(1), 97–125. http://doi.org/10.1146/annurev.environ.032108.105046

Rocchio, J. (1971). Relevance feedback in information retrieval, 313 – 323.

Rubin, G., Tohge, T., Matsuda, F., Saito, K., & Scheible, W. R. (2009). Members of the LBD family of transcription factors repress anthocyanin synthesis and affect additional nitrogen responses in Arabidopsis. *Plant Cell*, *21*(11), 3567–3584. http://doi.org/10.1105/tpc.109.067041

Ruffel, S., Krouk, G., Ristova, D., Shasha, D., Birnbaum, K. D., & Coruzzi, G. M. (2011). Nitrogen economics of root foraging: transitive closure of the nitrate-cytokinin relay and distinct systemic signaling for N supply vs. demand. *Proc Natl Acad Sci U S A*, *108*(45), 18524–18529. http://doi.org/10.1073/pnas.1108684108

Sadhukhan, A., Kobayashi, Y., Kobayashi, Y., Tokizawa, M., Yamamoto, Y. Y., Iuchi, S., … Sahoo, L. (2014). VuDREB2A, a novel DREB2-type transcription factor in the drought-tolerant legume cowpea, mediates DRE-dependent expression of stress-responsive genes and confers enhanced drought resistance in transgenic Arabidopsis. *Planta*, *240*(3), 645–64. http://doi.org/10.1007/s00425-014-2111-5

Sakamoto, H., Maruyama, K., Sakuma, Y., Meshi, T., Iwabuchi, M., Shinozaki, K., & Yamaguchi-Shinozaki, K. (2004). Arabidopsis Cys2/His2-type zinc-finger proteins function as transcription repressors under drought, cold, and high-salinity stress conditions. *Plant Physiology*, *136*(1), 2734–46. http://doi.org/10.1104/pp.104.046599

Sauter, A. (2001). The long-distance abscisic acid signal in the droughted plant: the fate of the hormone on its way from root to shoot. *Journal of Experimental Botany*, *52*(363), 1991–1997. http://doi.org/10.1093/jexbot/52.363.1991

Schmidt, R., Mieulet, D., Hubberten, H.-M., Obata, T., Hoefgen, R., Fernie, A. R., … Mueller-Roeber, B. (2013). Salt-responsive ERF1 regulates reactive oxygen species-dependent signaling during the initial response to salt stress in rice. *The Plant Cell*, *25*(6), 2115–31. http://doi.org/10.1105/tpc.113.113068

Selinski, J., & Scheibe, R. (2014). Lack of malate valve capacities lead to improved N-assimilation and growth in transgenic A. thaliana plants. *Plant Signaling & Behavior*, *9*(7), e29057. http://doi.org/10.4161/psb.29057

Seo, K.-I., Lee, J.-H., Nezames, C. D., Zhong, S., Song, E., Byun, M.-O., & Deng, X. W. (2014). ABD1 is an Arabidopsis DCAF substrate receptor for CUL4-DDB1-based E3 ligases that acts as a negative regulator of abscisic acid signaling. *The Plant Cell*, *26*(2), 695–711. http://doi.org/10.1105/tpc.113.119974

Shimazaki, H., & Shinomoto, S. (2010). Kernel bandwidth optimization in spike rate estimation. *Journal of Computational Neuroscience*, *29*(1-2), 171–182. http://doi.org/10.1007/s10827-009-0180-4

Smoot, M. E., Ono, K., Ruscheinski, J., Wang, P.-L., & Ideker, T. (2011). Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics (Oxford, England)*, *27*(3), 431–2. http://doi.org/10.1093/bioinformatics/btq675

Strogatz, S. H. (2001). Exploring complex networks. *Nature*, *410*(6825), 268–276. http://doi.org/10.1038/35065725

Stuart, J. M., Segal, E., Koller, D., & Kim, S. K. (2003). A Gene-Coexpression Network for Global Discovery of Conserved Genetic Modules. *Science*, *302*(5643), 249–255.

http://doi.org/10.1126/science.1087447

Takei, K., Ueda, N., Aoki, K., Kuromori, T., Hirayama, T., Shinozaki, K., … Sakakibara, H. (2004). AtIPT3 is a key determinant of nitrate-dependent cytokinin biosynthesis in Arabidopsis. *Plant Cell Physiol*, *45*(8), 1053–1062. http://doi.org/10.1093/pcp/pch119

Tanay, A., Sharan, R., & Shamir, R. (2005). Biclustering Algorithms: A Survey. In *In Handbook of Computational Molecular Biology Edited by: Aluru S. Chapman &amp; Hall/CRC Computer and Information Science Series*. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.133.9434

Tran, L.-S. P., Nakashima, K., Sakuma, Y., Simpson, S. D., Fujita, Y., Maruyama, K., … Yamaguchi-Shinozaki, K. (2004). Isolation and functional analysis of Arabidopsis stress-inducible NAC transcription factors that bind to a drought-responsive cis-element in the early responsive to dehydration stress 1 promoter. *The Plant Cell*, *16*(9), 2481–98. http://doi.org/10.1105/tpc.104.022699

Tripathy, B. C., Sherameti, I., & Oelmuller, R. (2010). Siroheme: an essential component for life on earth. *Plant Signal Behav*, *5*(1), 14–20. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/20592802

Valafar, F. (2002). Pattern Recognition Techniques in Microarray Data Analysis. *Annals of the New York Academy of Sciences*, *980*(TECHNIQUES IN BIOINFORMATICS AND MEDICAL INFORMATICS), 41–64. http://doi.org/10.1111/j.1749-6632.2002.tb04888.x

Vandepoele, K., Quimbaya, M., Casneuf, T., De Veylder, L., de Peer, Y., & Van de Peer, Y. (2009). Unraveling Transcriptional Control in Arabidopsis Using cis-Regulatory Elements and Coexpression Networks. *Plant Physiology*, *150*(2), 535–546. http://doi.org/10.1104/pp.109.136028

Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, *7*(1). http://doi.org/10.1186/1471-2105-7-91

Vidal, A. E., Álvarez, J. M., Moyano, T. C., & Gutiérrez, R. A. (2015). Transcriptional networks in the nitrate response of Arabidopsis thaliana. *Current Opinions in Plant*

Biology, *In Press*.

Walker, M. G., Volkmuth, W., Sprinzak, E., Hodgson, D., & Klingler, T. (1999). Prediction of Gene Function by Genome-Scale Expression Analysis: Prostate Cancer-Associated Genes. *Genome Research*, *9*(12), 1198–1203. http://doi.org/10.1101/gr.9.12.1198

Wang, X., Han, T. X., & Yan, S. (2009). An HOG-LBP human detector with partial occlusion handling. In *Computer Vision, 2009 IEEE 12th International Conference on* (pp. 32–39). Kyoto: IEEE. http://doi.org/10.1109/ICCV.2009.5459207

Wilkinson, S., & Davies, W. J. (2002). ABA-based chemical signalling: the co-ordination of responses to stress in plants. *Plant, Cell and Environment*, *25*(2), 195–210. http://doi.org/10.1046/j.0016-8025.2001.00824.x

Wright, D. P., Huppe, H. C., & Turpin, D. H. (1997). In Vivo and in Vitro Studies of Glucose-6-Phosphate Dehydrogenase from Barley Root Plastids in Relation to Reductant Supply for NO2- Assimilation. *Plant Physiol*, *114*(4), 1413–1419. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/12223780

Yang, Z. R. (2004). Biological applications of support vector machines. *Briefings in Bioinformatics*, *5*(4), 328–338. http://doi.org/10.1093/bib/5.4.328

Youngs, N., Penfold-Brown, D., Bonneau, R., & Shasha, D. (2014). Negative example selection for protein function prediction: the NoGO database. *PLoS Computational Biology*, *10*(6), e1003644. http://doi.org/10.1371/journal.pcbi.1003644

Yu, H., Kim, P. M., Sprecher, E., Trifonov, V., & Gerstein, M. (2007). The importance of bottlenecks in protein networks: correlation with gene essentiality and expression dynamics. *PLoS Computational Biology*, *3*(4), e59. http://doi.org/10.1371/journal.pcbi.0030059

Zhang, W., Morris, Q., Chang, R., Shai, O., Bakowski, M., Mitsakakis, N., … Kooy, D. van der. (2004). The functional landscape of mouse gene expression. *Journal of Biology*, *3*(5), 21+. http://doi.org/10.1186/jbiol16

Zhao, X.-M., Chen, L., & Aihara, K. (2008). Protein function prediction with high-throughput data. *Amino Acids*, *35*(3), 517–530. http://doi.org/10.1007/s00726-008-

0077-y

Zuberi, K., Franz, M., Rodriguez, H., Montojo, J., Lopes, C. T., Bader, G. D., & Morris, Q. (2013). GeneMANIA prediction server 2013 update. *Nucleic Acids Research*, *41*(Web Server issue), W115–22. http://doi.org/10.1093/nar/gkt533