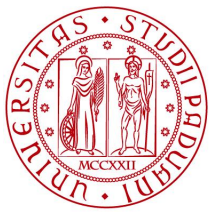

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede amministrativa: Università degli Studi di Padova

Dipartimento di Biologia

SCUOLA DI DOTTORATO: BIOSCIENZE E BIOTECNOLOGIE
INDIRIZZO: GENETICA E BIOLOGIA MOLECOLARE DELLO SVILUPPO
CICLO: XVIII

INTEGRATING GENE EXPRESSION DATA TO INFER HOW BIOLOGICAL CHANGES DRIVE TRANSCRIPTIONAL RESPONSES

Direttore della scuola: Ch.mo Prof. Paolo Bernardi

Coordinatore d'indirizzo: Ch.mo Prof. Rodolfo Costa

Supervisore: Ch.ma Prof.ssa Chiara Romualdi

Co-supervisore: Ch.mo Dott. Kristof Engelen

Dottorando: Marco Moretto

A Sara, Alice e Francesco

Acknowledgements

Arrivati alla fine di un lungo percorso, come è quello di dottorato, generalmente la lista di persone verso cui si ha un debito di riconoscenza è considerevolmente lunga. Il mio caso non fa eccezione. In primo luogo ringrazio la mia tutor di dottorato, Chiara Romualdi per i consigli, il supporto e i momenti di sana goliardia che non sono mancati. Un grazie particolare anche ai loschi figuri di cui il gruppo di Chiara è composto, ovvero: Gabriele Sales, Enrica Calura, Paolo Martini e Valentina Cappelletti con cui ho condiviso dal primo giorno questa avventura. I colleghi (ed ex-colleghi) dell'Unità di Biologia Computazionale in Fondazione. Paolo Sonego, Davide Albanese, Samantha Riccadonna, Pietro Franceschi, Federico Vaggi, Luca Bianco, Paolo Fontana, Andrea Cattani, Paolo Francesco Lenti e Claudio Donati, che rendono FEM un luogo di lavoro piacevole e stimolante a cui devo un numero x di piaceri e da cui, in generale, ho imparato molto (quando presto attenzione :-P). Alessandro Cestaro che, assieme a Paolo Fontana, è stato il mio primo vero tutor e che tutt'ora tutoreggia (ti devo una doccia Ale). Tutti i colleghi del Dipartimento di Genomica e Biologia della piante da frutto, tra cui Fabrizio Costa ed in particolare Mirko Moser, con cui si condivide volentieri oltre al *grande* ufficio anche una birra e qualche partita a *Descent*. Mario Di Guardo verso cui nutro sentimenti altalenanti ma che correlano con la preparazione della *granatina* siciliana ed il suo ostinato talento per il genere neo-melodico. Un grazie ai miei genitori, mia sorella, Andrea, Edoardo e Lorenzo. Distanti ma sempre presenti. Un ringraziamento particolare va a Carla e Valerio. Senza di voi non so davvero dove saremmo e, soprattutto, se questa tesi avrebbe mai visto la luce. Un enorme grazie ovviamente va a Sara, che ogni giorno mi onora della sua compagnia e da cui traggo (sfinendola) la forza necessaria per ogni progetto, grande o piccolo che sia. Con te accanto, tutto diventa *possimpibile*[1]. Infine un sentito grazie al mio co-supervisore Kristof Engelen per il costante supporto, la pazienza e la mole

[1] `https://www.youtube.com/watch?v=GuLcxg5VGuo`

di tempo che mi ha dedicato per (tentare di) insegnarmi qualcosa. Senza il tuo aiuto difficilmente sarei riuscito a terminare (e probabilmente iniziare) questo lavoro di dottorato. Grazie davvero.

Contents

Riassunto

Questa tesi di dottorato tratta principalmente di due argomenti tra loro interconnessi: il primo è lo sviluppo di una serie di *tool* per l'integrazione di dati di espressione genica. Il secondo è lo sviluppo di metodologie per la modellazione matematica di tali dati. Nella prima parte, quindi, viene descritta la metodologia utilizzata per integrare dati di espressione genica disponibili nei principali *database* pubblici, la creazione di una serie di strumenti *software* che implementano tali metodologie e l'applicazione di quest'ultimi al fine di realizzare collezioni di dati di espressione (*compendia*) per diversi procarioti ed una specie eucariote di interesse agrario (*Vitis vinifera*). Tali *compendia* sono particolarmente rilevanti applicate alla *systems biology* in quanto forniscono una ricca fonte di informazione. Essi sono delle matrici di espressione in cui ogni riga rappresenta un gene della specie di interesse, mentre le colonne rappresentano le diverse condizioni in cui l'espressione genica è stata misurata. Oltre ad essere il risultato della prima parte di questo lavoro di dottorato, i *compendia* di espressione sono anche il punto di partenza per la seconda parte che ha lo scopo di facilitare l'interpretazione biologica dei dati attraverso inferenza su modelli matematici creati a partire da essi. In particolare vengono discussi e sviluppati due modelli tra loro complementari. Il primo utilizza un approccio Bayesiano modellando una distribuzione di probabilità sul vero cambiamento dell'espressione di un particolare gene in risposta ad una particolare condizione. Il secondo modello sfrutta le reti Booleane per modellare l'informazione strutturale dei meccanismi genetici noti di risposta agli stimoli. Le reti Booleane vengono utilizzate per la creazione di una distribuzione di probabilità sui possibili stati stazionari delle cellule presenti nel campione effettivamente misurato. Utilizzando questi modelli è possibile, ad esempio, formulare ipotesi statisticamente valide sugli stimoli/segnali maggiormente responsabili dell'espressione di alcuni geni, sulla innata variabilità di un determinato gene (indipendentemente dalle condizioni in cui esso è misurato) oppure trovare complessi schemi di co-espressione genica.

Abstract

The work presented in this Ph.D. thesis is two sided. The first part describes a series of tools to integrate gene expression data, while the second one describes how to mathematically model them. The first part explains the methodology used to integrate publicly available transcriptomic data, the creation of a series of software tools that implement this methodology, and their application to create collections of gene expression data (*compendia*) for several prokaryote species and one eukaryote (the crop plant *Vitis vinifera*). *Compendia* are gene expression matrices in which every row is a gene of the species of interest while columns represent the different conditions in which genes have been measured. They provide a rich source of information for systems biology applications. Besides being the result of the first part of this Ph.D. project, gene expression *compendia* are the starting point for the second part, with the purpose of facilitating biological knowledge discovery drawing inference from mathematical models. We develop and discuss two complementary models. The first one uses a Bayesian approach, in which we model a probability distribution over an underlying true change in expression for a given gene in response to a given condition. The second one uses Boolean networks to model structural information about the known genetic mechanisms of response to stimuli. Boolean networks are used to fit a distribution over steady-states of cells in measured samples. These models may be used for various types of statistical inference and decision making. They can serve to formulate statistically sound hypothesis about stimuli/signals that better explain observed changes in gene expression, or about the inherent variability of a gene (independently from the conditions in which it is measured), or to find complex patterns of co-expression.

CHAPTER 1

Introduction

> The "-ome, -omics" serves well the "low input, high throughput, no output" modern, nowadays science.
>
> Michael I. Lerman, Sydney Brenner

1.1 Background

Recent progress in high-throughput genomics, proteomics, and transcriptomics have transformed biology in an information-intensive science. The amount of data produced by the latest technologies is unprecedented and the need to use computational methods to manage, analyze, and interpret information is increasing. Bioinformatics and computational biology refer to interdisciplinary fields that use concepts from computer science, statistics, and engineering to analyze and interpret biological data. The National Institutes of Health (NIH) defines bioinformatics as "research, development, or application of computational tools and approaches for expanding the use of biological, medical, behavioral or health data, including those to acquire, store, organize, archive, analyze, or visualize such data" while computational biology is defined as "the development and application of data-analytical and theoretical methods, mathematical modeling and computational simulation techniques to the study of biological, behavioral, and social systems" [1]. Following these definitions, the project presented in this Ph.D. thesis is both a *bioinformatics* and a *computational biology* work. It is divided in two parts schematically represented in figure 1.1. The first one presents

a methodology that aims to "acquire, organize, analyze and visualize" gene expression data. Chapter 2 describes the implementation of the tools used for the creation of gene expression compendia, while chapters 3 and 4 describe the application of such gene expression compendia to prokaryotic and plant species respectively. The second part presents "the development and application of data-analytical and theoretical methods" to model such data and in particular section 5.1 describes a Bayesian framework to model gene expression data as presented in a compendium. Section 5.2 describes another approach that is based on previous Bayesian model and employs Boolean network to model gene expression data from a different perspective. Finally, the last chapter 6 presents a conclusion and discusses future perspectives.

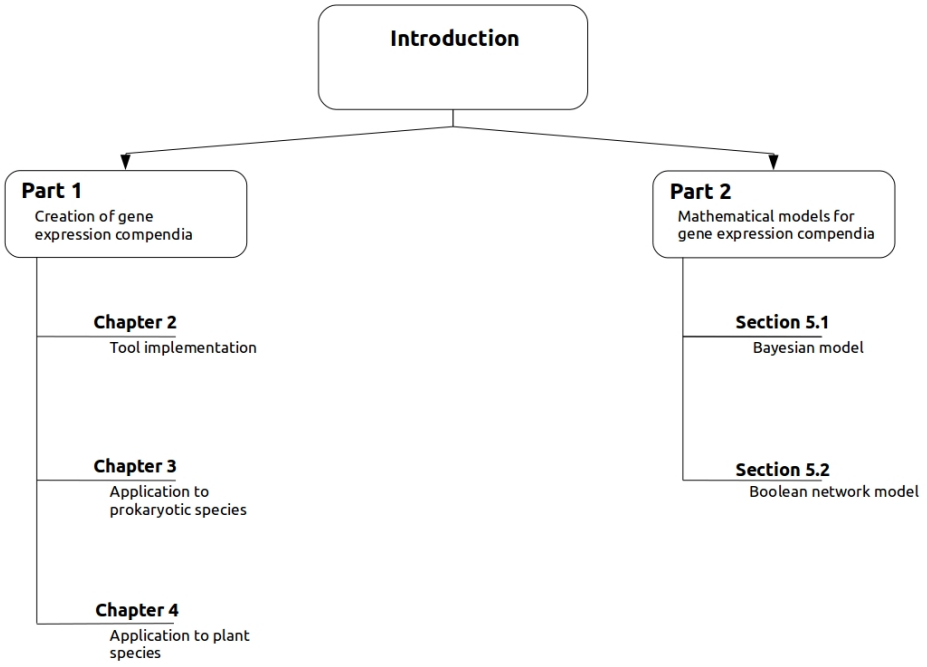

Figure 1.1: **Thesis organization** This thesis is divided into two parts. The first one describes the creation and application of gene expression compendia to prokaryotic and plant species. The second part is concerned with two mathematical approaches to model such data.

1.1.1 Transcriptomics and data integration

Transcriptomics is the study of the complete set of RNA transcripts produced in one cell or in a population of cells under specific environmental conditions, using high-throughput methods based on microarray or next-generation sequencing

technologies. The widespread use of these tools has resulted in a rapid accumulation of gene expression data in public repositories such as NCBI GEO [2], ArrayExpress [3] and NCBI SRA[1]. Such repositories have the enormous potential to provide an *holistic* view of how different experimental conditions leads to gene expression changes, by comparing transcriptome fluctuations across all possible measured conditions. Unfortunately, this is not a task easily achievable due to differences among laboratories and technology platforms that make direct comparisons difficult. Nonetheless, in recent years there have been several efforts to fulfill data integration of gene expression studies. One issue in data integration is technical variability due to different laboratories' working methods. To assess the agreement on experimental results, several initiatives [4], [5], [6] compared data obtained from different laboratories using microarray and RNA-seq platforms with identical RNA samples. Since there are several biological and technical sources of variability to be considered, and the employment of an advanced technology does not eliminate either, only a careful experimental design is effective in order to keep bias and batch effects under control. Proposed approaches to integrate gene expression analysis usually can be categorized as direct integration or meta-analysis: they can either directly consider the sample-level measurements within each study[7], and merge these into a single data set or select only some features assumed to be related across studies, such as parameters that capture the relationship between genes and phenotypes [8]. Direct integration tries to overcome the limits of meta-analysis with model-based approaches[12], that can directly integrate gene expression data and better account for confounding effects. This is generally done on experiments of the same platform, using e.g. the Robust Multi-Array averaging (RMA) [13], [14] normalization, because for direct integration of experiments from different experimental platforms one needs to adjust the data for batch effects (which are usually confounded with true biological changes) by e.g. a Bayesian framework. Meta-analysis integrates gene expression analysis combining information from primary statistics [9] (such as p-value) or secondary statistics [10] (such as gene list) resulting from single studies. Those studies manually combine the information from several data sources defining confidence levels subjectively for each individual study without a general scheme. Meta-analysis is a common method to integrate conclusions from different studies. Goldstein *et al.* [11] analyze several meta-analysis approaches used for combining results of independent studies discussing general pros and cons of the meta-analysis approach.

[1] http://www.ncbi.nlm.nih.gov/sra

1.1.2 The COLOMBOS approach to gene expression data integration

One of the efforts towards data integration in gene expression studies is COLOMBOS [15], originally COLlection Of Microarrays for Bacterial OrganismS, developed for three bacterial species (*Escherichia coli*, *Bacillus subtilis*, and *Salmonella enterica* serovar Typhimurium) and recently updated with sixteen others prokaryotic species [16], [17] and including also RNA-seq technology. COLOMBOS is a comprehensive organism-specific cross-platform expression database that provides a suite of tools for the exploration, analysis and visualization of gene expression data. COLOMBOS' approach to data integration is unique in the sense of directly combining gene expression information from different technological platforms and experiments. Data and experiment-related information (*meta-data*) are gathered and curated starting from raw intensities or sequence reads for microarrays and RNA-Seq respectively. A robust normalization and quality control procedure is performed to permit direct comparison of gene expression values across different experiments and platforms. This results in a single expression matrix in which each row represents a gene and each column represents a 'sample contrast'. Sample contrasts measure the difference between a *test* and a *reference* condition, both of which are extensively annotated with various sorts of meta-data. The expression data itself are log-ratios (base 2), so that positive values represent up-regulation, and negative values represent down-regulation of a gene in the test sample compared to the reference sample. COLOMBOS falls under the direct integration methodology, but without the need for batch-normalization as calculating logratios, for contrasts that are defined by samples that come from the same experiment and platform combination (a 'batch'), ensures that a lot of batch related variation is removed [18]. COLOMBOS principal goal is to gather together as many expression data as possible for a given organism to explore patterns of co-expression across several experimental conditions. The creation of a co-expressed genes cluster (known as *module*) is performed similarly to a BLAST [19] search in which COLOMBOS looks for expression values for a given set of conditions, but using expression correlation instead of sequence similarity to score the best matches. Modules can be modified in several ways in order to highlight their genes' behavior and to analyze (anti)co-expression patterns. COLOMBOS has shown to be a valid resource both as an exploratory tool and as a gene expression database used for downstream analysis [20], [21], [22],[23],[24],[25],[26],[27],[28],[29],[30].

1.2 An integrated and flexible environment for data acquisition

COLOMBOS technologies are composed by a front-end web application used to access and analyze gene expression data in the compendia and by a back-end

suite of tools, dubbed COMMAND (COMpendia MANagement Desktop), designed to facilitate the acquisition, standardization, annotation, pre-processing and homogenization of public expression data for compendia creation. The original COLOMBOS and COMMAND implementations date back to 2011 and the original code has been extensively modified to account for new functionalities added during the last years of development, eventually growing to the point of making it difficult to efficiently run and maintain the application. COMMAND quickly developed from a set of various scripts used to download, parse and analyze public gene expression data to a fully-functional web-application. Unfortunately, the rapid evolution brought some drawbacks, mainly:

- **performance problems**, due to the overhead caused by the use of several layers of abstraction to the data model and the use of different programming languages and;

- **lack of flexibility** in data acquisition, due to the high number of *ad hoc* code written to include specific experiment designs. Unfortunately this led to the paradoxical situation where at times it's easier to modify original data than the code itself to account for the uniqueness of experiment data formats.

In order to tackle both problems we completely overhauled the COMMAND implementation trying to remove all the bottlenecks starting from the adoption of a unique programming language (Python) used for data acquisition, data presentation and mathematical calculation. Moreover, a unique point-of-access to the database data-model has been created using a Object Relational Mapping (ORM) software layer. Finally, an extremely flexible and powerful tool has been implemented in order to manage any possible situation that could possibly arise during data acquisition.

COMPASS

Since COMMAND is a complex program, one of the main concerns for the transition from the old implementation to the new one, was the need to keep the former still active during the development of the latter for the time necessary to have a working prototype. To this purpose we developed a software layer, called COMPASS (COMPendia Applications Support Structure) that implement basic functionalities shared between COMMAND and COLOMBOS applications and abstract the current database data-model for it to be still usable from the old version but decoupled from the new implementation in order to be easily changed to accommodate any future needs. Each newly implemented functionality dismantles one or more of the old implementation in a stepwise phase-out fashion in order to guarantee to always have a working environment.

COMMAND

The new implementation of the back end program for data acquisition is based on the assumption that it would be worthless to try to manage every possible way in which public expression data are deposited. Instead, we provide a powerful tool to manage experiment uniqueness shaping the experiment structure and injecting user-defined Python scripts to mine for experiment primary information and raw data. Moreover, the new implementation provides several tools to create an expression compendia from scratch and to manage both raw and annotation data.

COLOMBOS application to plant species

COLOMBOS had been originally developed to collect microarray data for prokaryotes. Over the years it evolved allowing both the integration of different transcriptomics technologies, like RNA-seq, and the creation of compendia for archeae and eukaryotes. While collecting a large amount of data for model organisms, like *Escherichia coli*, is facilitated due to the great number of experiments performed, for non-model species the situation is usually pretty different as only few experiments are available. The importance of expression data integration in this case is even more significant given the need for an adequate magnitude of data to be able to draw valid and general conclusions. The application of COLOMBOS technology to grapevine species led to the development of VESPUCCI (Vitis Expression Studies Platform Using COLOMBOS Compendia Instances)[31], a gene expression compendium that include most of publicly available expression data for grapevine. Working with a non-model plant species highlighted the need to significantly rethink some aspects of the data acquisition and annotation process. The creation of a gene expression compendium using COLOMBOS technology is made easier thanks to the aid provided by the COMMAND tool but it is still mainly a manual effort. The peculiarity and complexity of both plant transcriptome and experiment design required the possibility to flexibly manage how micorarray probes and RNA-Seq short read sequences are mapped and thus assigned to a measurable gene. The same concept of 'measurable transcipt' was also used to account for some technical limitations, like the impossibility to distinguish among some genes given the high sequence similarity they share.

1.3 Mathematical models for gene expression compendia

COLOMBOS first and foremost goal is to bring together as much data as possible, and opening this comprehensive data up for exploration and search for complex (co)expression patterns. While COLOMBOS provides a rich resource for top-down systems biology or for complementing more focused molecular

biology research, one of the drawbacks is that the potential for rigorous statistical inference is not used. Simply relying on existing tools is unfeasible, due to the cross-experiment/cross-platform nature of the data, the complications associated with varying number of replicate sample contrasts, the existence of self-self contrasts (measuring only biological variability), and the issue of dependence through a shared reference sample. To overcome this limitation and further extend the range of usage of COLOMBOS, we developed a statistical framework that can be employed for various types of inference and decision making, explicitly taking into account dependencies between contrasts and working irrespective of the number of replicate sample contrasts available. It serves as a basis for 'interrogating' the data in a statistically sound way to answer diverse questions such as: identifying differentially expressed genes for one (or more) contrast(s), finding complex patterns of co-expression, classification/prediction and 'biomarker' discovery. While the purpose of statistical framework is to provide a sound statistical model to deal with data in the compendia, a different approach, that exploits the statistical model, was developed to describe how our knowledge of the 'system' (i.e. which genetic entities have the potential to interact or be involved in related biological processes) can explain the observed genome-wide expression responses to a 'stimulus' or a shift in biological conditions. This second approach extrapolates a single-cell model to population level in order to account for how gene expression measurements have been collected during experiments and relationships among genes.

1.3.1 A Bayesian noise model

Using a Bayesian approach we developed a statistical framework that model a probability distribution over the underlying true change in expression for a gene in response to a shift in biological conditions. We defined the probability distribution:

$$p(\mu_x|X, G, C)$$

where μ_x is the underlying true change in gene expression for the given gene G, in response to a given contrast C, with $X = (x_1, \ldots, x_n)^T$ being the n replicate expression log-ratios. Such an approach has several advantages that are particularly relevant given the requirements:

- inferring the **complete posterior** $p(\mu_x|X, G, C)$ distribution, instead of using point estimators, gives more flexibility with respect to the kind of questions we would like to answer;
- the inherent **sequential** nature of Bayesian learning makes it well-suited for the disparateness in the number of replicates present in the compendia;
- the Bayesian formulation provides a convenient way for introducing **prior knowledge**, such as the dependence that exists between contrasts shar-

ing the same references as well as general properties of the data and its distribution for gene G and contrast C that we know empirically.

1.3.2 Modeling contrasts with Boolean networks

Genes known to be involved in a biological process are represented as nodes (that can be either *on* or *off*) in a Boolean network, while the relationship among those genes are represented as Boolean functions. Expression data in the compendia are used to fit a model that uses Boolean network attractor states to *simulate* the different steady-states in which sub-populations of cells were at the time measurements were taken. The fitted estimated weights of each possible attractor state represent the proportion of cells in a particular steady-state during the shift in biological conditions.

Part one

CHAPTER 2

Creating gene expression compendia

2.1 Compendium creation workflow

The typical workflow for compendia creation via COMMAND is described in the original COLOMBOS paper [15] (here we report and briefly describe it for the sake of completeness). There are three main steps toward the creation of a gene expression compendium using COLOMBOS technology. Each of them requires a set of dedicated tools and they have to be done sequentially.

Collection of gene expression data

Public repositories such as NCBI GEO ([2]) and ArrayExpress ([3]) are accessed through the available Application Programming Interface (API). All the microarray experiment information together with the related platform information are downloaded. Raw data (or normalized data when raw aren't available) are stored in a unified format. Microarray probe sequences are also stored and mapped in a platform-specific way to a unique list of genes that is composed by the organism's RefSeq file available at NCBI. These genes correspond to the rows in the final expression matrix. If probes aren't available, it's gene target is identified by other information such as the locus tags or common gene names. Regarding RNA-Seq experiments, pre-processing such as quality control, cleaning of raw reads and alignment on a reference genome (or transcriptome) is done separately. Raw counts associated with the organism specific gene list are stored together with the related platform information.

Annotation of samples and experiments

After raw intensities and platform-related information have been stored in the database, the next phase constitutes the definition and annotation of condition contrasts. As stated in the introduction (section 1.1.2) a 'condition contrast' does not represent a single experimental condition, but rather the difference (in log-ratios) between a *test* and a *reference* condition . Samples are tagged as 'test' or 'reference' for single channel microarray experiments and RNA-Seq experiments, while for dual channel microarray experiments, usually one of every two array hybridizations serves as a reference to the other. If a sample does not represent a unique and distinctive biological condition (such as samples of genomic DNA or a pool of different samples) it is not considered and gets discarded. This choice ensures the biological interpretability of every defined contrast, as the associated log-ratios measure a change in expression in response to quantifiable stimuli that have been altered from the reference to the test sample. After contrasts are defined, they are annotated using terms from a controlled vocabulary created to ensure both computational tractability and human readability. The creation of the controlled vocabulary is a manual process, in which new terms are neatly added to the vocabulary tree as needed during the importing of experiment samples.

Homogenization of gene expression data

The last step in the creation of the compendium is the homogenization of gene expression data. Various pre-processing techniques are carried on in order to render experiments from different platforms comparable. The following 'general rules' are applied whenever possible during this step:

- raw intensities/reads are preferred over already pre-processed data as data source;
- no local background correction or mismatch probe correction are performed;
- non-linear normalization techniques are performed;
- variance dispersion stabilization is performed on RNA-seq data.

2.2 Implementation of a workbench for compendia creation

COLOMBOS is an extensive program. It's main characteristic is given by the possibility of exploring a huge database of gene expression data scanning for patterns of (anti)co-expression. In spite of his apparent complexity it 'only' has to deal with one well-defined data-model providing a user-friendly interface

to easily browse the expression matrix. The real complexity lies in the creation of such expression matrix, i.e. in the back-end program used to collect, annotate, and manage gene expression data. The COMpendia MANagement Desktop (COMMAND) has been created with the purpose of simplifying the three necessary steps to create a compendia (as explained above in section 2). The original COMMAND code was composed by several scripts, mainly written in the Matlab and Ruby programming languages that have grown in time, up to the point of being considered a fully functional application. The step towards the creation of a web-application had been completed by adding PHP and JavaScript code to glue together all those scripts. Its evolution from a 'bunch of scripts' to a fully-featured application that 'just works' has been fast and more time had been dedicated to adding new features instead of correcting old mistakes. Because of that, the COMMAND source code had grown to the point of being hard to be further developed and maintained. The main problems with the old implementation of COMMAND are:

- **The use of several programming languages**: Ruby, PHP and Matlab have been used to code the server side part of the application. This has lead to redundancy (especially between some Ruby and PHP parts), a general lack of performance, and difficulty in debugging, maintaining and deploying the code given the need to install all code dependencies as external packages.

- **More points of access to the database**: all languages independently access to the database. This leads once again to redundancy (in saving credentials for accessing the database) and general difficulty in understanding which part of the code accesses the database to retrieve (or store) data and passes it to another part of the code.

- **Unused functionalities**: some features have been developed but never fully exploited. Such design choices have to be taken into account when dealing with the data-model weighing down the typical workflow.

- **Some persistent data are stored in files**: not all critical data are stored in the database, some of them are organized in files and directories. This has several drawbacks such as the lack of control for integrity with database data, the difficulty in retrieving them since the location is sometimes hidden in the file-system, and the general lack of security given the possibility to accidentally delete them.

- **Data-model abstraction using XML files**: this is linked with 'unused functionalities' as extra code has been developed with the purpose of ensuring enough abstraction and scalability in case of changes in the data-model. Unfortunately, the drawbacks overpower the advantages as the overhead given by the intermediate creation of XML files slows down the whole execution.

- **Lack of flexibility**: a lot of code has been developed to account for the uniqueness and heterogeneity of specific data-formats and experimental designs. This created several problems during data acquisition as new experiments not always match previously developed code used for other, similar experiments. This sometimes leads to the contra-productive situation in which modifying original data formats, instead of the code to handle them, was the easiest strategy to import data.

2.3 COMMAND: revisited

Thanks to the newly available technologies, measuring the complete transcriptome is an easily achievable task nowadays. New experiments and data becomes publicly available on a daily basis and thus a tool like COMMAND is particularly useful in order not only to build gene expression compendia but also to keep them up-to-date. Unfortunately, updating compendia using the old COMMAND implementation soon became unfeasible given the lack of flexibility needed to correctly manage and import gene expression experiments as each of them requires a specific way to be correctly handled and imported. The decision to re-implement COMMAND has been made focusing on some major improvements that had to be carried out, such as:

1. overhaul the way in which experiments are imported;
2. update the database structure;
3. create a coherent server-side programming interface;
4. use less programming languages (possibly one);
5. update the client-side software libraries (ExtJS);

In order to successfully fulfill both the need for a coherent server-side programming interface and the usage of possibly only one programming language to be used as server-side web-application, general business logic programming and numerical calculation, the choice naturally fell on Python. Python is a versatile programming language thanks to the plethora of freely available modules and libraries developed from the community. There's also several working environments for Python that greatly simplify and speed up the development and debugging of Python applications.

COMMAND implementation

Figure 2.1 shows the old and the new COMMAND implementation. They are both centered around the database structure that holds the data-model used from both COMMAND and COLOMBOS applications. Each application (like COMMAND and COLOMBOS) are composed by a client-side, that is the GUI

(Graphical User Interface) developed in Javascript using the ExtJS framework, and a server-side developed in Python using the Django Framework. ExtJS is a JavaScript application framework for building interactive cross-platform web applications using Ajax (Asynchronous JavaScript And XML). ExtJS includes several GUI controls, such as text field, grid control, tabs, etc. . . , to be used within web applications. It greatly simplifies the development of user interfaces since it already provides most of the widgets ready to be plugged-in and used. Django[32] is a high-level Python web framework that helps the development of web applications since it automatically deals with most of the issues during development of such applications. In the new version COMPASS, COMMAND and COLOMBOS are technically all Django applications. COMPASS has been developed as a software library that deal with the database, using the Object Relational Mapping (ORM) provided by Django, and expose an API to the other applications. The end result is that every application that needs to retrieve or store data from the database, won't need to access to it directly but instead would instantiate Python Objects that represent tables in the database. Since Django takes care about most of the details of a web-application, the flow of execution and the communication between client and server-side is pretty basic. Essentially for every *event* (like pressing a button or ordering a table grid) a request through an Ajax call is made from the user interface invoking a Python function. The function (which is part of a coherent application interface) takes care of the response possibly invoking specific functionalities implemented in the COMPASS library. The exchange of information between client and server side (that is parameter passing and response object) is done using JSON (JavaScript Object Notation) objects.

2.4 Gene expression data collection workflow

Gene expression data are represented and deposited upon publication as *experiments*. An *experiment* is a collection of *samples* measured on a *platform* (typically one platform per experiment, but sometimes more than one are used). A sample is a genome-wide measurement that represents the RNA abundances of all genes expressed in a given condition, while a platform is the technology used to perform the actual measurement. As already stated above, the steps needed to create a compendia are three, but the first one (i.e. data collection) is by far the most involved and is the only one that has been radically revisited, while the annotation and homogenization steps are practically left unchanged at this point in time (phase-out development to retain functionalities). The complexity in data collection arises from the disparateness of ways in which public expression data are made available. Since there's little to no control on data format and content, experiments are essentially all different from one another. In order to deal with such heterogeneity in data formats, we implemented a procedure composed by three steps:

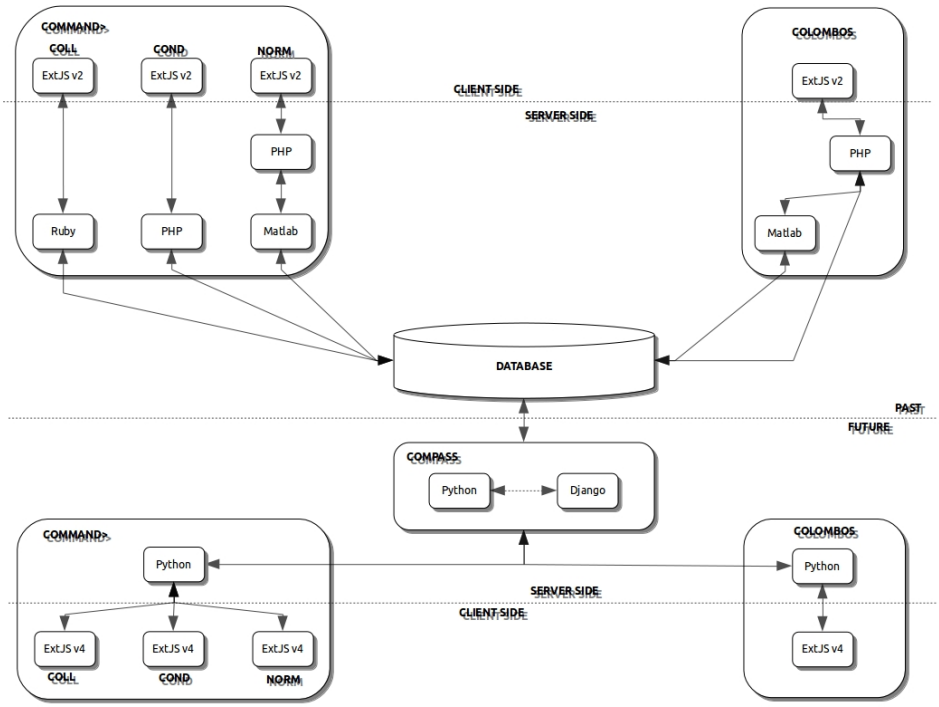

Figure 2.1: **Past and future software layer implementation** The upper part of the figure shows the old implementation with several point of access to the database and all the different programming languages used. The lower part shows the new implementation with COMPASS as the only software layer that deal with the database and implements the data-model while each of the applications have one coherent Python interface to manage all server-side functionalities.

- experiment selection and raw-data download;
- experiment structure definition;
- experiment parsing and data extraction.

Figure 2.2 shows the interface for the first step. This first part allow to see all public gene expression experiments from GEO and ArrayExpress. From here is possible to select and download data for the new experiments we wish to import, or specify which experiments we explicitly want to exclude. From this same interface it is also possible to manage already imported or partially imported experiments, together with platform information (description, probe sequence, etc. . .) and gene information (sequence and functional annotation). According to the different steps of the import process in which an experiment can be, it is labelled with a different status:

- **searched**: the experiment has been added to the list of experiments to be imported from the search result (first part of the import process);
- **structured**: the experiment structure has been defined (second part of the import process);
- **parsing**: parsing scripts have been assigned to experiment files (third part of the import process);
- **included**: the experiment is imported in the database;
- **annotated**: the experiment is annotated;
- **excluded**: the experiment is of no interest and won't be imported.

The second part of the import process (see Figure 2.3) is done defining the experiment structure, i.e. the platforms and the samples the experiment is composed of, as a hierarchy tree. In this context, a *platform* is defined as the specific technological platform used to measure RNA abundance (for example an Affymetrix chip), with all the associated information and meta-information like probes sequences or name and description of the platform. A *sample* is intended as all the information and meta-information, such as raw measurements, name and description, related to a single biological RNA sample measured with a specific platform. (Note that a single biological RNA sample is not limited to a single gene, but consists of all transcripts of the genes that are expressed in that sample.) Once this is done, files that contains meta-information, data and measurements are associated with the respective experiment, platform and sample in the hierarchy. This approach is completely different from the original one and it provides great flexibility because of the way in which we can assign any file to any entity in the hierarchical structure of the experiment in order to get out the relevant information.

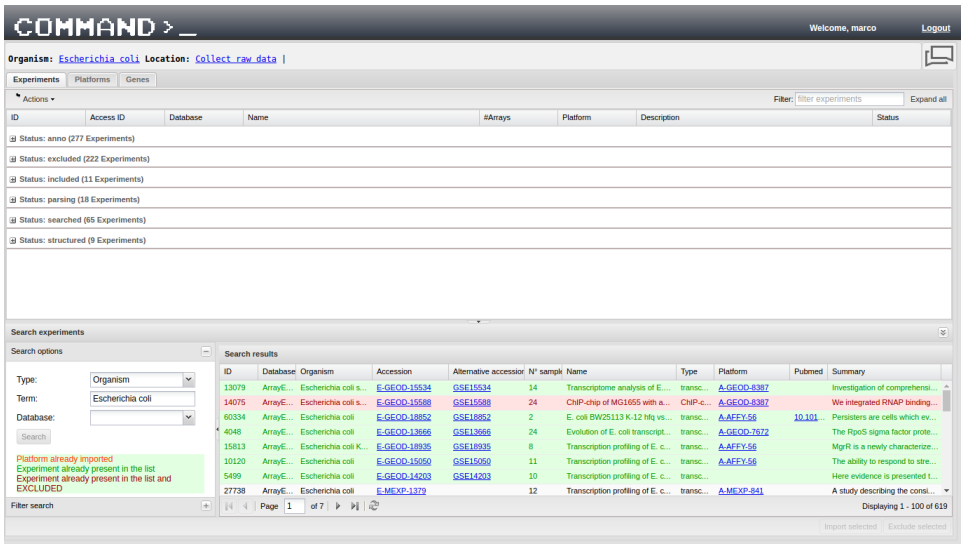

Figure 2.2: **Experiment selection.** The top grid includes all experiments already part of the compendia and experiments already imported but not completely processed. The bottom part is another grid that shows all the experiments found on public databases for a given query (using the search panel on the bottom left part). Any words defined by the user can be used as search terms. Different colors allow to easily recognized experiments already imported or experiments different from gene expression then won't be imported.

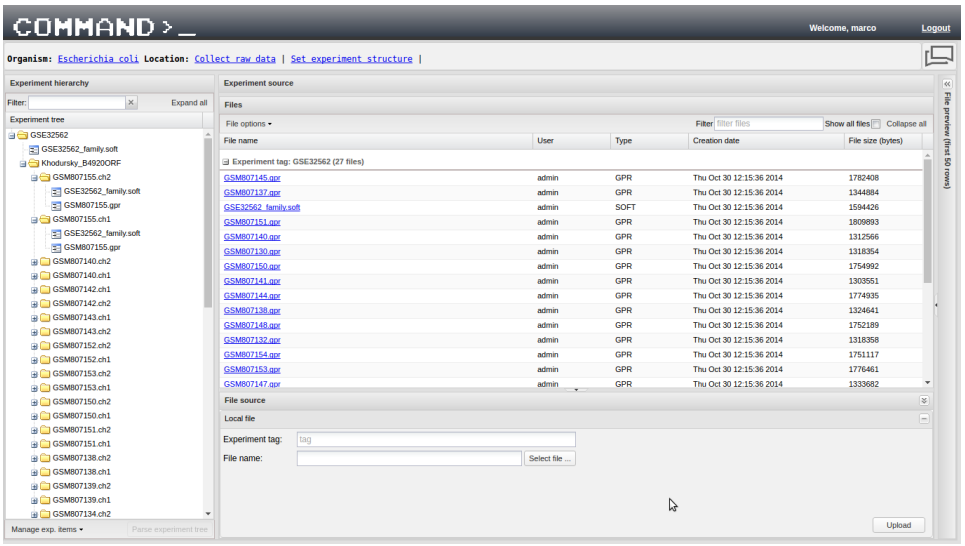

Figure 2.3: **Experiment structure definition.** The right-hand panel shows all the downloaded files associated with the experiment (it is also possible to manually upload files). The tree on the left side is the experiment hierarchy (experiment, platforms and samples) with the respective files associated. All these steps can be performed manually or let COMMAND perform it automatically.

The third and last step (see Figure 2.4) for experiment raw data import comprises the association of Python scripts to each of files associated with experiments, platform and samples. Those Python scripts parse and extract information from files and populate an internal data structure (a Python object that represent the whole experiment) with all the necessary data. The possibility to use custom Python script to parse out the information from raw files gives all the flexibility needed to cope with any situation that could possibly arise during experiment import. On the other hand this could expose the server to potentially harmful Python code. For this reason COMMAND contains an administration panel to handle users and groups of users and to set detailed rights and privileges to data access and the use of functionalities. Before data are actually imported a series of checks are performed in order to verify completeness and integrity of data. Once all the checks are satisfied the experiment can finally be imported into the database.

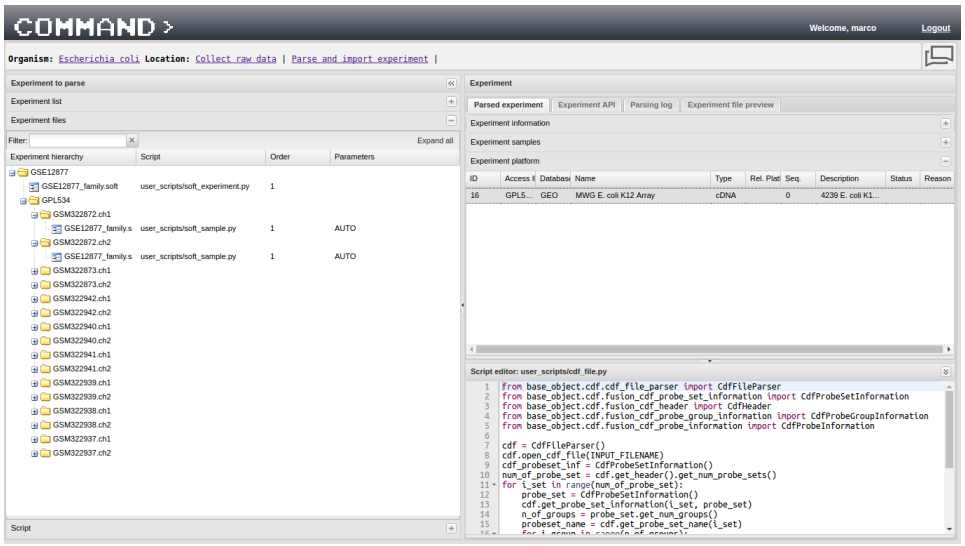

Figure 2.4: **Parsing and importing experiment.** Experiment structure defined in the previous step is shown on the left side panel. A Python script can be associated to each file (together with arguments if needed) that would be executed (following a particular order if needed) filling the *experiment object*. Available Python scripts are listed in a (hidden) panel below the experiment hierarchy. On the bottom-right side there's an editor to create and modify Python scripts. The top-right panel contains a preview of the *experiment object* divided in three tabs: experiment, platform and sample. Each of them shows currently data extracted from raw files through the execution of Python scripts.

CHAPTER 3

COLOMBOS v3.0: leveraging gene expression compendia for cross-species analyses

Marco Moretto, Paolo Sonego, Nicolas Dierckxsens, Matteo Brilli, Luca Bianco, Daniela Ledezma-Tejeida, Socorro Gama-Castro, Marco Galardini, Chiara Romualdi, Kris Laukens, Julio Collado-Vides, Pieter Meysman and Kristof Engelen[17]

Abstract

COLOMBOS is a database that integrates publicly available transcriptomics data for several prokaryotic model organisms. Compared to the previous version it has more than doubled in size, both in terms of species and data available. The manually curated condition annotation has been overhauled as well, giving more complete information about samples' experimental conditions and their differences. Functionality-wise cross-species analyses now enable users to analyse expression data for all species simultaneously, and identify candidate genes with evolutionary conserved expression behaviour. All the expression-based query tools have undergone a substantial improvement, overcoming the limit of enforced co-expression data retrieval and instead enabling the return of more complex patterns of expression behaviour. COLOMBOS is freely available through a web application at `http://colombos.net/`. The complete database is also accessible via REST API or downloadable as tab-delimited text files.

Introduction

COLOMBOS is a collection of expression data from both microarray and RNA-Seq experiments for several prokaryotic species, taken from publicly available database such as the Gene Expression Omnibus (GEO) [2] and ArrayExpress [3]. Its uniqueness resides in the ability to cope with data heterogeneity and directly integrate data coming from different platforms and technologies. Other gene expression compendia are usually built either from data for a single transcriptomics platform or they rely on the integration of expression analysis results, rather than the integration of the actual measurements. In COLOMBOS however, data are collected and curated starting from the original raw intensities for microarrays and sequence reads for RNA-Seq, and then processed with a robust normalization and quality control pipeline to allow direct comparison of gene expression behaviour across different experiments and platforms [15]. This results in a single expression matrix for every species, its rows representing the measured genes and its columns representing condition contrasts, comparisons between test and reference samples of different biological conditions. Attention is also given to the acquisition of metadata related to the description of the biological conditions surveyed in an experiment, so that all the included samples and condition contrasts are formally annotated by means of a controlled vocabulary of condition properties. This annotation is a manual effort with the purpose of making the data comparable from a biological viewpoint and to yield reliable interpretations of expression patterns. COLOMBOS compendia are accessible using the web interface, through a set of REST API calls, or via the R [33] package Rcolombos; they are also available for download in their entirety for use of COLOMBOS data in third-party stand-alone applications. Different types of analyses can be done using the COLOMBOS web interface itself; typical operations include starting from a set of known genes to find the conditions where they are (co)-expressed or to identify additional co-expressed genes. COLOMBOS' tools are designed for users to 'play around' with the compendia, exploring the data with respect to the biological question they are interested in. They are encouraged to try different types of search queries based on genes or conditions, the available annotations or by relying on the actual expression values in a way reminiscent of a BLAST functionality with gene expression behaviour instead of sequence similarity. They can then visualize their results, use them as a basis for new queries to find additional (anti-)co-expressed genes, generate clusters to separate disjoint expression profiles, explore the overlap between multiple query results and potentially combine them, etc. There are several detailed use case tutorials on the website, illustrating step-by-step how concrete examples of conceptually different biological questions could be handled through the COLOMBOS interface. The previous v2.0, with all of its original databases and tools, will be kept available for future reference along side COLOMBOS v3.0; how to access it is explained in the website's Help section.

Data content update

COLOMBOS v2.0 [16] was composed of seven bacterial species, four more than it contained at its inception. The current update includes an additional twelve species of biomedical or industrial relevance, including some Archaea. The main criteria for selecting these new species were the amount of publicly available expression data and quality of genome annotation and their perceived status as model organisms. A complete overview of the available species and associated statistics can be found in Table 3. The previous compendia have also been updated with recent experiments, in some extreme cases leading to an almost 2-fold increase of available data. For instance, the biggest compendium is that for *Escherichia coli*, which now contains over 4000 condition contrasts, nearly 2000 more than COLOMBOS v2.0 and almost as many as its number of genes, rendering the expression matrix virtually square. Gene lists, representing the species' measurable transcripts, have been created from the NCBI RefSeq database [34] and various gene annotation data were added (or updated) from UniProt-GOA [35], RegulonDB [36], BioCyc [37] and EcoCyc [38], or species-specific published datasets [21].

Complete sample annotation

COLOMBOS sports an annotation system for condition contrast related meta-data which relies on a manually curated and controlled vocabulary. It is an essential information source that aids in the interpretation of gene expression patterns. As COLOMBOS condition contrasts represent comparisons between two samples (a 'test' sample compared to a 'reference' sample), in the past only condition properties which represented actual differences between the two samples were annotated. The major drawback of this approach is that it disregards what is shared between both samples: two contrasts could be annotated exactly the same regardless of the condition 'background' of their individual samples. For instance, when two contrasts had measured the exact same decrease in oxygen concentration, they would have been annotated identically. If one of the contrasts however had wild-type strains for both test and reference samples, and the other contrast had strains with a mutation in a gene important in aerobic respiration, this information would not be apparent from the contrast's annotation, while it is arguably an important factor to acknowledge. For this COLOMBOS update, we have fully overhauled the annotation system to instead work at the sample level (as opposed to the contrast level) and consequently hold the meta-information for both a contrast's samples' experimental conditions, and not only the differences between them. When looking up a condition contrast in the COLOMBOS database, you will now be presented with the biological background (e.g. strains, medium, growth conditions) as well as the biological difference that results in the displayed expression behaviour.

	Strain	Number of genes	Number of contrasts	Missing values (%)	First inclusion	Samples	Experiments	Platforms
Escherichia coli	MG1655	4321	4077	3.6	v1.0	5510	254 [15]	73
Bacillus subtilis	168	4176	1259	3.7	v1.0	1814	45	35
Salmonella enterica serovar Typhimurium	cross-strain	6261	1066	41.6	v1.0	1856	36	22
Streptomyces coelicolor	LT2	4556	172	6.4	v2.0	316	8	10
Pseudomonas aeruginosa	14028S	5416	681	22.7	v2.0	1252	17	7
Helicobacter pylori	SL1344	4655	213	9.8	v2.0	288	11	9
Bacillus anthracis	A3(2)	8239	371	7.3	v2.0	546	7 [2]	7
Bacillus cereus	PAO1	5647	559	1.6	v2.0	592	33	2
Bacteroides thetaiotaomicron	26695	1616	133	3.1	v2.0	256	8	5
Campylobacter jejuni	Ames	5039	66	3.0	v3.0	75	4	4
Clostridium acetobutylicum	ATCC 14579	5231	283	2.4	v3.0	392	16	10
Lactobacillus rhamnosus	VPI-5482	4816	333	1.9	v3.0	353	19	4
Methanococcus maripaludis	NCTC 11168	1572	152	12.5	v3.0	260	14	11
Shigella flexneri	ATCC 824	3778	377	2.4	v3.0	419	12	11
Sinorhizobium meliloti	GG	2834	79	3.6	v3.0	158	3	2
Streptococcus pneumoniae	S2	1722	364	1.5	v3.0	728	19	3
Thermus thermophilus	301	4315	35	17.0	v3.0	38	3	3
Yersinia pestis	1021	6218	424	2.7	v3.0	713	20 [19]	10

Table 3.1: Rows of the table represent all the species and strains for which a gene expression compendium is hosted. Columns represent (from left to right): the species name, the strain used as reference genome for microarray probe to gene mapping and RNA-Seq read alignment, the total number of genes in the compendium, the total number of contrasts in the compendium, the percentage of missing values, the COLOMBOS version of the first inclusion of the respective species or strain, the total number of samples from which the compendium's contrasts are built, the total number of corresponding experiments on GEO and ArrayExpress (the latter indicated between square brackets) and the total number of platforms represented.

Functionality update

Cross-species analysis

A completely new functionality in COLOMBOS v3.0 is the ability to work with all species simultaneously. The data from different organisms have been integrated on a higher level based on clusters of homologous genes (CHG) constructed with OrthoMCL v2.0.9 [39] using the default settings as applied to the protein sequences for the strains included in COLOMBOS v3.0. These CHGs can be thought of as the rows of an overarching expression matrix obtained by stitching together the individual compendia. Expression data for orthologous genes, i.e. genes assigned to the same CHG, are aligned across the respective species; species without a representative gene in a CHG can be thought of as having missing values. In case a CHG contains paralogous genes (multiple genes from the same species), their expression values are averaged. All data analysis tools included in COLOMBOS have been adapted to deal with these new cross-species compendia, so that this complex expression matrix can be queried and explored with the same flexibility as any single species. The cross-species comparison is not only a novelty for the identification of co-expressed gene sets across several species for e.g. evolutionary studies, but also has several advantages for the way compendia can be constructed. We can now build compendia for different strains and integrate them at the species level using homologue mappings. This has a clear advantage as, instead of using a single reference strain's genome to represent the species as was done before, we can now explicitly recognize genomic differences between strains and thus improve read alignment (RNA-seq) or probe to gene mapping (microarrays) to generate higher quality expression data. This concept has been used to improve our *Salmonella enterica sp. Typhimurium* compendium, where the original consisted of more or less equal parts of three different strains with minor differences in their genomic content.

Analysis tools

Several changes have been made to web portal's suite of analysis tools and the RESTful web service and R API. These are mainly related to the query functionalities that actually make use of the expression values themselves ('BLASTing with expression data'). While these previously looked solely for consistent co-expression, they are now capable of returning complex patterns of expression behaviour across sets of query genes (or conditions). For instance, in v2.0 the Quicksearch functionality would return, for a set of user defined genes, the contrasts where those genes behave in a similar and coherent way. These are not necessarily the most informative, or relevant, contrasts for the user, especially for larger gene sets for which co-expression behaviour might be rare and unrepresentative. By default the Quicksearch in v3.0 will visualize complex patterns of co-expression by running a biclustering on the returned module data, and

will not necessarily return contrasts where the query input genes behave in the same way (although this functionality is still available in the Advanced search). Other improvements include various export functionalities so that COLOMBOS results can be easily imported in other widely used tools or databases (such as Cytoscape [40], BioCyc) for further downstream analysis.

Discussion and future plans

COLOMBOS' growths over the years have been a continuous effort towards better gene expression data integration and easier exploration and interpretation. Not only has the data more than doubled, but this last major update is another step in the direction of improving the strengths and eliminating the weaknesses of the previous version(s). The redesigned condition annotation system provides a more reliable interpretation of expression patterns with respect to the biological stimuli that are causing them. The new cross-species capabilities have the obvious advantage over the old system to be able to perform gene expression analyses on all species simultaneously, but also enable more accurate measurements mapping by separating different strains within the same species. Keeping the compendia up-to-date, as well as expanding the scope by adding new organisms, is naturally our first priority. We generally select new species or strains based on data availability, but are always open to suggestions or requests from users who are interested in access to a gene expression compendium for a particular species. Further improvements and new functionalities that revolve around cross-species capabilities are planned for future versions. Flexibility regarding CHGs selection and composition, as well as new tools to empower users when dealing with complex CHGs are amongst the priorities. For instance, instead of being limited to pre-calculated, fixed CHGs for which homologues cannot be re-defined and that encompass all species in the compendia as is the case now, users will be able to define the settings to create CHGs for the species of their choice and consequently more dynamically integrate the data from the corresponding compendia. Updated tools will likewise enable a finer management of CHGs, unlike e.g. the current paralogues' expression calculation that is averaged across all paralogues without the possibility for a different evaluation considering the variability amongst those paralogues, as well as give users the ability to compare expression derived measures, such as co-expression scores or networks, across species.

CHAPTER 4

VESPUCCI: exploring patterns of gene expression in grapevine

Marco Moretto, Paolo Sonego, Stefania Pilati, Giulia Malacarne, Laura Costantini, Lukasz Grzeskowiak, Giorgia Bagagli, Maria Stella Grando, Claudio Moser and Kristof Engelen[31]

Abstract

Large-scale transcriptional studies aim to decipher the dynamic cellular responses to a stimulus, like different environmental conditions. In the era of high-throughput omics biology, the most used technologies for these purposes are microarray and RNA-Seq, whose data are usually required to be deposited in public repositories upon publication. Such repositories have the enormous potential to provide a comprehensive view of how different experimental conditions lead to expression changes, by comparing gene expression across all possible measured conditions. Unfortunately, this task is greatly impaired by differences among experimental platforms that make direct comparisons difficult. In this paper, we present the Vitis Expression Studies Platform Using COLOMBOS Compendia Instances (VESPUCCI), a gene expression compendium for grapevine which was built by adapting an approach originally developed for bacteria, and show how it can be used to investigate complex gene expression patterns. We integrated nearly all publicly available microarray and RNA-Seq expression data: 1608 gene expression samples from 10 different technological platforms. Each sample has been manually annotated using a controlled vocabulary developed ad hoc to ensure both human readability and computational tractability. Expression data in the compendium can be visually explored using several tools provided by the

web interface or can be programmatically accessed using the REST interface. VESPUCCI is freely accessible at `http://vespucci.colombos.fmach.it`.

Introduction

Grapevine (*Vitis* spp.) is an economically important fruit crop and one of the most cultivated crops worldwide [41]. Grape berries are consumed as fresh fruit or used for high-valued commodities as wine or spirits. Grapevine transcriptomics studies started over a decade ago, initially using microarrays but later, exploiting the sequenced genomes [42], [43] and the availability of high-throughput sequencing, also using RNA-Seq approaches. As system biology becomes more prevailing in everyday analysis, one of the pressing aspect of analysis is how to integrate different sources of information into one coherent framework that can be interrogated in order to gain knowledge about the system as a whole [44]. Prior to biological information integration across several levels (such as proteomics, transcriptomics, and metabolomics), it is important to acquire and combine all the possible available information within each specific field. Together with the methodological problem of combining different sources of information, there's the more practical issue of having sufficient data to justify data integration in the first place, because in order to draw general and valid conclusions a large amount of data is a desirable feature. While for model species this is hardly an issue, for non-model crop species the number of performed experiments might be limited, the technological platforms less established, and heterogeneous data a further complicating factor. Nevertheless, as biology is turning into a data-driven science the prospect of large dataset availability becomes more and more feasible even for non-model species, and in terms of gene expression and functional analysis there have been several efforts to fulfill data integration in different organisms including grapevine [45], [46], strawberry [47], and citrus [48]. In this paper, we present an expansive grapevine gene expression compendium that can be used to analyze grapevine gene expression at a broad level. It was created based on an approach for dealing with the large heterogeneity of data formats present in public databases, and to integrate cross-platform gene expression experiments in one dedicated, coherent database. The proof-of-concept of this approach was presented in [15] as a web-application for exploring and analyzing specific expression data of several bacterial species. This original technology platform has already been used as a basic framework for creating a gene expression compendium for a more complex case as the multicellular, higher eukaryote Zea mays [49]. Here, we used the most updated version of the COLOMBOS technology [17] to show how this approach can be further extended for the creation of gene expression compendia on other important crop species, focusing our attention on grapevine gene expression studies. Regardless of the available tools, most of the steps toward the creation of such a compendium, require a massive amount of manual

Platform name	Platform type	Number of samples
NimbleGen 090918 vitus vinifera exp HX12	Microarray	583
Affymetrix V. vinifera (grape) genome array	Microarray	502
Affymetrix GrapeGen V. vinifera GrapeGena520510F	Microarray	219
INRA V. vinifera oligo array 15K v3	Microarray	100
Combimatrix GrapeArray 1.2	Microarray	69
Illumina HiSeq 1000	RNA-seq	60
Illumina HiSeq 2500	RNA-seq	36
AB 5500 xl genetic analyzer	RNA-seq	20
Illumina HiSeq 2000	RNA-seq	12
Illumina genome analyzer IIx	RNA-seq	7

Table 4.1: Overview of all samples imported in VESPUCCI ordered by number of samples. The first column contains the name of the transcriptomics platform, the second column is the type of platform either microarray or RNA-Seq. The third column contains the number of samples measured with the respective platform imported in VESPUCCI.

curation, from defining a controlled vocabulary for description of experimental conditions to the interpretation of experiment designs and annotation of the included samples. The benefits of Vitis expression studies platform using COLOMBOS compendia instances (VESPUCCI) lie in the availability of the whole known measured transcriptome activity of grapevine in a single programmatically accessible repository and the possibility to extensively explore gene expression patterns through the visual tools made available by the web interface.

Materials and methods

Data sources

The experiments included in VESPUCCI have been collected from the Gene Expression Omnibus [2], ArrayExpress [3], and the Sequence Read Archive (SRA)[1]. The majority is made up of microarray experiments (91% of samples), with the 'NimbleGen 090918 Vitus HX12 array' and 'Illumina HiSeq 1000' being the most used platforms among microarray and RNA-Seq experiments, respectively. Table 4.1 shows the summary of samples imported per platform. The complete overview of imported experiments and platforms is available in Supplementary Table S1.

[1] `http://www.ncbi.nlm.nih.gov/sra`

Gene annotation

The CRIBI V1 gene prediction[2] and associated sequences for *Vitis vinifera* PN40024 (cv. Pinot Noir) have been used as the base gene transcript list. Corresponding gene functional annotations have also been added. Together with the original CRIBI annotation, which comprises GO [50], KEGG [51], Pfam [52], ProSite [53], and Smart [54], the VitisNet [55] molecular network was also included.

Sample annotation

Samples in VESPUCCI have been manually curated using a controlled vocabulary to precisely describe which parameters have changed across different experimental conditions. The creation of the controlled vocabulary is an ongoing adaptive manual process, in which curators add or modify new terms as needed during the acquisition of new experiment samples, keeping the vocabulary as concise and organized as possible. Terms in the vocabulary have largely been introduced ex novo following the original experimental designs, but on occasion have also been borrowed from other annotation systems like the Plant Ontology[3] [56] for describing the plant anatomical structures or the modified Eichhorn–Lorenz scale [57] for describing grapevine-specific developmental stages. The complete vocabulary, along with its hierarchical structure, is available in the Supplementary Table S2.

Compendium creation

The compendium creation process can be divided in three major steps: data collection and parsing, sample annotation, and data homogenization. To facilitate these three steps and to deal with the complexity of maintaining big amounts of data and meta-data, we have relied mostly on the COLOMBOS v2.0 [16] and v3.0 [17] backend managing applications. For this *V. vinifera* expression compendium, new tools were added to the COLOMBOS backend software, mainly related to the probe-to-gene (re)mapping. Specifically, microarray probes are now aligned by a two-step filtering procedure using the BLAST+ program [19]. The two filtering steps are done to ensure that probes not only map to genes with high similarity (restrictive alignment threshold), but also that they map uniquely (unambiguously) to a single location and be less prone to cross-hybridization (less restrictive alignment threshold). Probes of different microarray platforms generally vary in terms of length, species/cultivar of origin, and sequence quality. To always obtain the reasonably best possible alignment according to each platform's specific characteristics, parameters, and cutoff thresholds were employed on a platform-specific basis.

[2] `http://genomes.cribi.unipd.it/DATA/V1/`
[3] `http://www.plantontology.org/`

Results

Vitis vinifera gene expression compendium

At the core of the VESPUCCI *V. vinifera* compendium is a gene expression matrix that combines publicly available transcriptome experiments from the most common microarray and RNA-Seq platforms (an overview is given in Table 4.1 and Supplementary Table S1). VESPUCCI's distinctive characteristics are its data integration strategy and the way in which it handles information coming from different platforms and technologies, which is based on COLOMBOS technology. Data and meta-data are gathered and curated starting from raw intensities or sequence reads for microarrays and RNA-Seq, respectively. A robust normalization and quality control procedure is performed to permit direct comparison of gene expression values across different experiments and platforms. This results in a single expression matrix in which each row represents a gene and each column represents a 'sample contrast'. Sample contrasts measure the difference between a 'test' and a 'reference' sample from the same experiment. The decision as to which samples are paired to form contrasts, is made in part based on technical considerations as explained in [15], and in part on the desire to deviate as little as possible from the original intent of the experiment. Both samples, and the differences between them, are then extensively annotated with various sorts of meta-data. The expression data itself are log-ratios (in base 2), so that positive values represent up-regulation, and negative values represent down-regulation of a gene in the test sample compared to the reference sample. VESPUCCI's compendium was built with specific modifications and additions for *V. vinifera* to the COLOMBOS technology, and these are described in the following sections.

Defining measurable gene transcripts

The list of measurable gene transcripts, representing the rows of the expression matrix, is based on the CRIBI V1 gene annotation, with some modifications to optimize probe-to-gene remapping (see next section), and read alignment. An important consideration for this remapping is that the CRIBI V1 gene predictions can show (regions of) high similarity, which is not uncommon for plant crop species. As a result, probes can end up matching perfectly, or near perfectly, to more than one gene. According to the way in which, we built the compendium, such shared, ambiguous probes would usually be discarded because of their inability to reliably measure one single gene. Instead of removing these probes, with consequent loss of information, we decided to keep them as a measurement of a whole cluster of genes, implying those genes expression changes can only be assessed as a whole but not individually. The decision is a trade-off between losing probes (measurements) and losing the possibility to distinctively measure each gene as a single entity. We used the Nimblegen

platform to investigate both ambiguous probes behavior and gene prediction structure, and decided on 466 cases in which genes can be 'clustered' together according to their sequence similarity and the probes they share. One clear-cut case to present the complexity of the issue is depicted in Figure 4.1. From this example is clear that each gene is actually measured on average by four probes (as expected) but, except for three probes (VitusP00165181, VitusP00165231, and VitusP00165171) all the other probes align perfectly (or near perfectly) to other genes, making impossible to distinguish one gene from another. In particular these four genes, beside being different among each other, are all annotated as Myb-related, a well-known transcription factor gene family composed by 100s of genes [58] and are positioned one after the other across chromosome 2 in a region of approximately 130 kb. This target cross-talk is corroborated by the actual probe-level intensities, which are highly correlated across all sample contrasts included in the compendium (Figure 4.2).

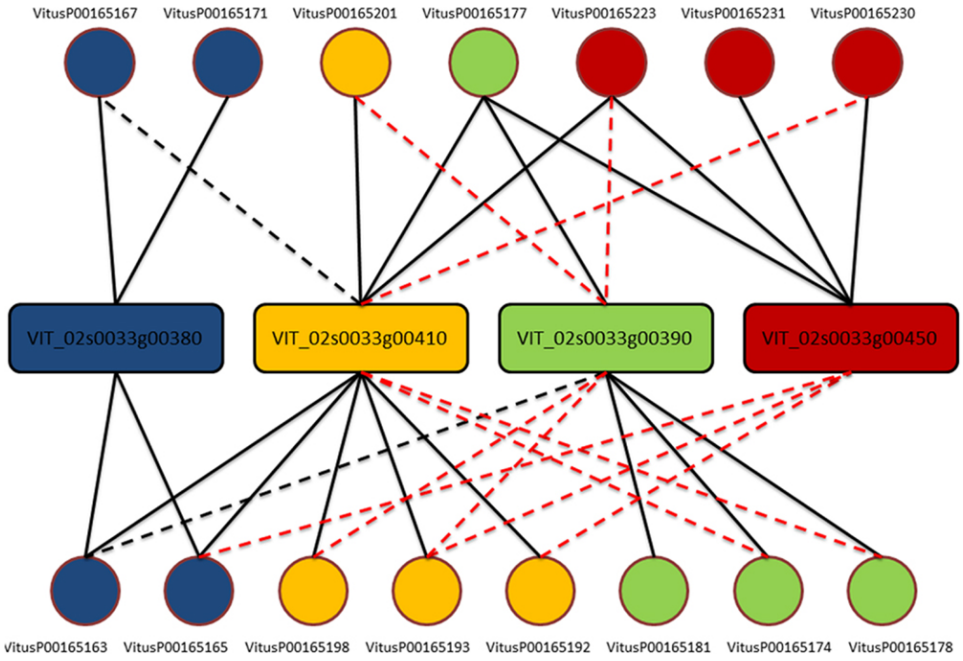

Figure 4.1: **Probe-to-gene mapping for cluster 170.** Genes (in rectangles) are colored accordingly to probes (circles) based on the original platform mapping. Each line corresponds to an alignment of the whole probe against one gene. A solid line means no mismatches, a black dashed line means one mismatch while a red dashed line means two or three mismatches.

To better understand the nature of gene-probe clusters, we carried out a survey of each of the 466 clusters. They consist in total of 1366 genes and 3472 probes, distributed across clusters as depicted in Figure 4.3. We inspected the clusters based on the probe-to-gene alignment quality and probe-level expression

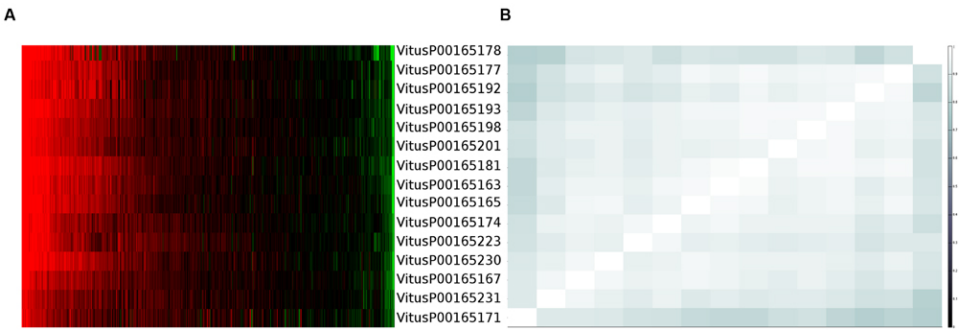

Figure 4.2: **Probe expression values and correlation for cluster 170.** (A) Probes expression values measured across more than 500 Nimblegen sample contrasts sorted by values. (B) Probes correlation matrix using uncentered Pearson correlation.

values across all Nimblegen experiments imported in VESPUCCI (38% of sample contrasts). The great majority of clusters consist of only a few genes with consistent behavior (according to probe expression patterns) and that are part of gene families and positioned one after the other along the same chromosome (or predicted on un-anchored loci). Other clusters are extremely dense and highly connected (e.g., clusters 1, 15, 176, and 177). Another set of clusters is composed by weakly connected genes (few probes) positioned on different chromosomes. For example cluster 283 is composed by five putative kinase proteins that span four chromosomes, and for which probes might be designed on a conserved catalytic domain. Some clusters present a 'perfect ambiguity' structure (e.g., clusters 47, 65) for which each probe aligns perfectly to each gene, making impossible to distinguish across measured genes. Interestingly, clusters with a non-perfect alignment (e.g., clusters 134, 220) instead show how probe level expression values reflect alignment mismatches, exposing the issue of measuring genomic variability instead of expression changes. Cases such clusters 185, 213, and others suggest that the measured genes could be allelic variants of the same gene as they are 99% similar with similar structure and predicted on contiguous or un-anchored loci. Finally, few other clusters appear to be problematic due to bad expression data and ambiguous probe-to-gene alignment (e.g., clusters 20, 21, and 42). All of the gene cluster related information (probe-to-gene alignment graphs, probe-level expression, and correlation heatmaps) is available as Supplementary Materials.

Probe-to-gene remapping

To take full advantage of an updated gene annotation and for a more coherent integration of different platforms, we remapped probes for each microarray platform to the CRIBI V1 gene prediction. Such remapping of probes to transcripts

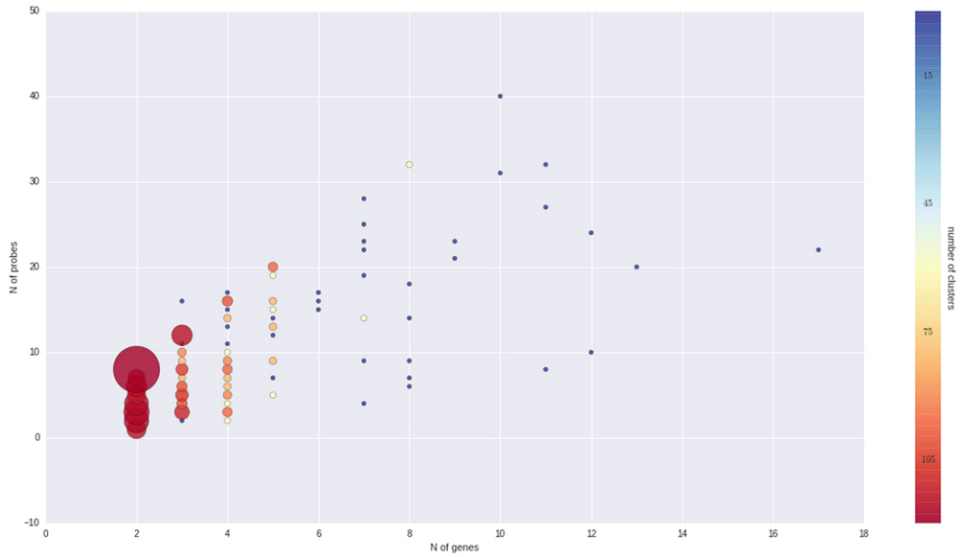

Figure 4.3: **Overview of gene clusters.** Both the size and color of the spheres are proportional to the number of clusters that is made up of a given number of genes and probes. It is clear that the great majority of clusters are composed by just few genes and probes.

has advantages over original annotations [59]. Different microarray platforms have different probe-to-gene alignment qualities. Given the disparateness in terms of number of samples, number of measured transcripts, and probe-to-gene mapping quality not all the available microarray platforms have been imported. The top performing platform is the Nimblegen microarray that shows a nearly perfect correspondence to the one in VESPUCCI. This is easily explained by the fact that it contains 118015 probes of 60 nucleotides with an average of four probes per gene and was specifically designed to match the CRIBI V1 gene prediction. It measures the expression of 29549 (out of 29971) gene predictions representing ∼98.6% of the genes of the CRIBI V1 gene prediction and 19091 random probes as negative controls [60], [61]. On the other hand, platforms like the 'University of Arizona Vitis buds spotted DNA/cDNA array' exhibit quite poor performance in terms of number of measured transcripts, probe-to-gene mapping, and probe signal (data not shown), which made us decide to exclude it from the compendium. The low quality can be ascribed to the fact that its 10369 probes have been designed from ESTs of two *V. vinifera* cultivars (Perlette and Superior) as well as the *V. riparia* species, and have an average length of nearly 1 kb. We compared our probe-to-gene mapping results to the original mappings for the microarray platforms using the complete gene annotation[4]

[4] `http://www.sdstate.edu/ps/research/vitis/pathways.cfm`

[62]. The results are reported in Table 4.2. The mapping is quite consistent to the original mappings, with the notable exception being the 'Combimatrix GrapeArray 1.2' platform, which lacks nearly 40% of correspondence between the mapped genes. The higher numbers for our mapping can be attributed to a different mapping program and strategy used, while the differences in overlapping gene mappings in the INRA and Combimatrix arrays could be due to the need of double mapping the probeset to the corresponding tentative consensus (TC) and then to the CRIBI V1 gene prediction in the gene annotation file. This could lead to two different gene ids if the genes are similar to each other or if the TC has been wrongly annotated.

Sample annotation

The *V. vinifera* gene expression compendium in VESPUCCI comes with an expansive and curated annotation of the biological conditions for all the included samples. Each sample in the compendium has been manually annotated using qualitative and quantitative terms from a controlled vocabulary specifically created for *V. Vinifera* (more information can be found in the Section 'Materials and Methods' and Supplementary Table S2). Annotating test and reference samples to conveniently show the differences and similarities between these samples provides a useful way to assess which are the potential driving properties responsible for the observed changes in expression. The condition annotation system, with its hierarchical vocabulary, provides a broad view of publicly available grapevine gene expression studies and the nature of the experiments that have been carried out (Figure 4.4). Nearly half of the VESPUCCI compendium is composed of sample contrasts measuring changes in developmental stages, particularly in the berry around véraison (Eichhorn–Lorenz stage 33–38), which is by far the most investigated topic. Together with development-related traits, biotic, and abiotic treatments also comprise a big chunk of available experiments. They include a variety of infections with several grapevine pathogens, together with temperature, water, and salinity stresses among others, while the preferred sampled tissue is fruit, as a whole or as separated parts, e.g., skin and flesh.

Vitis Expression Studies Platform Using COLOMBOS Compendia Instances (VESPUCCI)

The VESPUCCI web application is a specifically designed interface to interact with the expression data, without the need for external tools or programmatic skills. It is built around the idea of expression modules. A module is a subset of the whole gene expression matrix composed by rows and columns that represent genes and sample contrasts, respectively. A set of built-in tools serves for creation and modification of modules by querying the database for genes and sample contrasts in several ways. Users can look for expression patterns starting

Platform name	Original mapping	VESPUCCI mapping	Overlap	Missing values
NimbleGen 090918 vitus vinifera exp HX12	28811	29061	28069	3.7%
Affymetrix V. vinifera (grape) genome array	8581	9873	7954	66%
Affymetrix GrapeGen V. vinifera GrapeGena520510F	12593	13385	12200	53.9%
INRA V. vinifera oligo array 15K v3	6153	6582	4795	77.3%
Combimatrix GrapeArray 1.2	8956	9193	5448	69.5%

Table 4.2: Total number of genes measured per platform. First column contains the microarray platform name. The second column holds the number of measured genes according to the platform original probe-to-gene mapping. The third column contains the number of measured genes according to VESPUCCI probe-to-gene mapping. The fourth column contains the number of overlapping genes between the two mappings. The last column contains the percentage of genes for which there is no measurement.

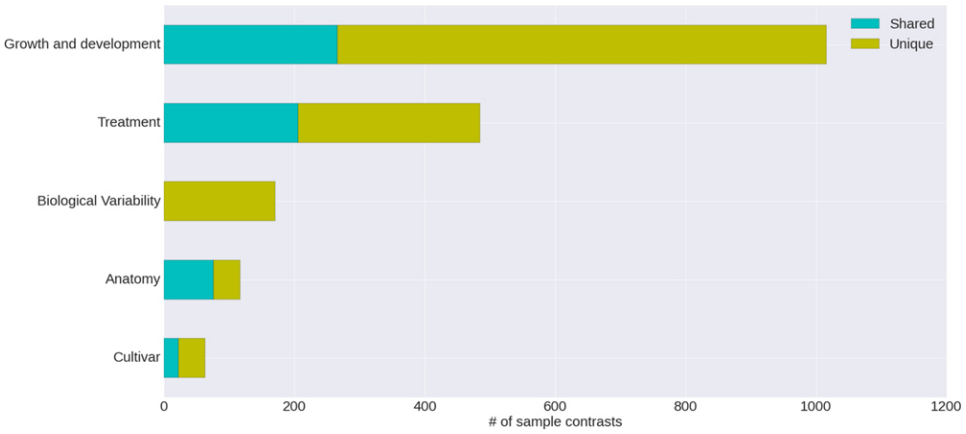

Figure 4.4: **Categories of annotated sample contrasts.** Number of sample contrasts annotated as measuring a change in one of five major categories. The differences between test and reference sample for some contrasts are related to more than one category; the proportion of these is indicated as 'shared' versus 'unique.'

from specific genes, conditions or whole experiments they are most interested in and extend or reduce expression modules with more genes or sample contrasts either manually or automatically relying on VESPUCCI's clustering algorithm. Similar to a BLAST search, VESPUCCI tries to retrieve expression values for a given set of conditions, but using expression correlation instead of sequence similarity to score the best matches. Alongside tools for building and modifying modules, the web interface comprises several tools to convey information, like annotation term enrichment, the correlation network and the complete contrast annotation that display the link between changes in biological condition and gene expression. The VESPUCCI compendium is also accessible through a set of REST API calls, or from within the statistical software environment R [33] via the R package Rcolombos. The web application of VESPUCCI is very much an exploratory tool to help researchers explore patterns of gene expression behavior for genes of interest. A prototype of VESPUCCI (dubbed MARCOPAOLO) has already been used to identify candidate genes involved in the fine regulation of anthocyanin and flavonol biosynthesis. In particular, co-expression with genes involved in the regulation of flavonoid biosynthesis was one of the criteria adopted to refine the list of genes identified in the genomic regions deduced by a QTL analysis for anthocyanin and flavonol content in ripe berries [63], [64]. A co-expression analysis against VESPUCCI was also carried out to find putative interacting partners and target genes of VvibZIPC22, one of the candidate genes specifically associated to flavonol biosynthesis, which is being proposed as a new regulator of flavonoid biosynthesis in grapevine. While both these cases represent a 'guilt-by-association' co-expression analysis, VESPUCCI's tools are

not limited to that and are designed to encourage users to play around with data in the compendium given the biological process they are interested in. One could also query for experiments of interest instead of genes, or simply study the behavior of (a set of) genes of interest across the different biological conditions without necessarily looking for other co-expressed genes. For instance, the top part of Figure 4.5 shows the results of a default Quicksearch for the 11 genes of the carotenoid cleavage dioxygenases (CCD/NCED) gene family, part of the grape carotenoid pathway [65]. The results of such a default search do not show all condition contrasts in the compendium, but only the top most relevant for the query genes, and can already provide insights into their behavior. First and foremost, it appears that the genes of this small gene family are not at all expressed in the same manner, and that for this particular family similarities in expression profiles are correlated up to a certain extent with the phylogenetic relationships between its genes [the superimposed tree in the bottom part of the figure is adapted from the phylogeny presented in Figure 6 of [66]]. A deeper inspection of that behavior not only confirms previously reported results, such as up-regulation at berry ripening of CCD4a and CCD4b, but not CCD4c [67], but it also provides some novel, potentially interesting leads for further exploration. For instance, there is a prominent -but not consistent- anti-correlation of NCED2 and NCED3 with CCD4a and CCD4b. There are also strong changes in expression of some gene family members in response to Eutypa lata infection. These sort of observations generally only represent the initial starting point of further VESPUCCI analyses, such as investigating these genes' behavior in other infection processes contained in the compendium, or maybe looking for co-expressed genes with NCED2/NCED3 or CCD4a/CCD4b. For an in-depth illustration of these concepts, we have included another case study in the website as well, which is presented there as a detailed step-by-step tutorial with the ability to load associated data directly in the interface. This particular case study is meant to show off VESPUCCI's most common features and capabilities in a hands-on manner. It focuses on a set of genes found to be modulated by the phytohormone abscisic acid (ABA) in pre-véraison berries (Stefania Pilati, personal communication); this list of genes was used as input to query the database. After performing any database query, VESPUCCI creates an expression 'module', a subset of the whole expression compendium determined by a set of genes and a set of sample contrasts and the corresponding expression values. The returned gene expression module indicated that the 55 ABA genes appear highly modulated in 353 experimental conditions in the VESPUCCI compendium. The default visualization of this module ('by expression'; Figure 4.6) emphasizes the interesting patterns of condition-dependent (anti-)co-expression behavior among this set of ABA genes. The gene annotation enrichment in turn reports their involvement in the response to stress and ABA, as well as in galactose metabolism. The main biological processes represented in our module, correspond to different biological contexts in which ABA affects gene expression: fruit and berry development, bud development,

and water and salinity stress. The explorative purpose of the web-interface is strengthened by tools used to modify the module by extending (or shrinking) it with new genes and/or contrasts. Continuing the analysis, the module was split according to these three biological processes, and these sub-modules then formed the basis for new queries to include additional genes with highly similar (or opposite) expression profiles in these three specific biological contexts. The final lists of (anti)-co-expressed genes are candidates for being involved in the pathways regulated by ABA, and/or for sharing similar, but currently unannotated mechanisms of regulation with the genes in the module.

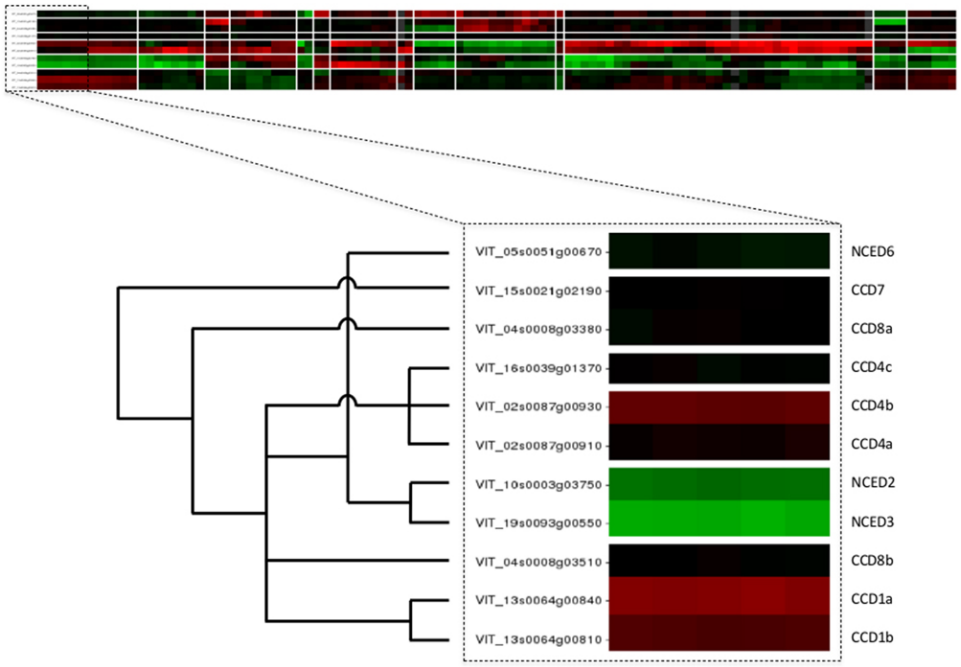

Figure 4.5: **Case study of carotenoid cleavage dioxygenases gene family.** The top part of the figure shows the VESPUCCI Quicksearch result for the 11 genes of the carotenoid cleavage dioxygenases (CCD/NCED), while the bottom depicts the superimposed phylogeny adapted from [66].

Discussion

In this paper, we present VESPUCCI, a gene expression compendium for grapevine that integrates publicly available transcriptomics data from several microarray and RNA-Seq platforms into one coherent database, queryable via a web or REST interface. The web interface is meant to be intuitive and flexible for non-expert users, and is designed to encourage them to 'play around' with the data

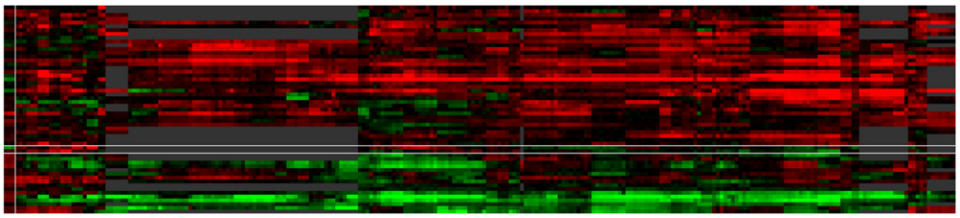

Figure 4.6: **Case study of ABA modulated genes.** The default 'by expression' visualization of VESPUCCI orders both genes and contrasts in this heatmap (resp. rows and columns) in such a way as to highlight the different patterns of condition-dependent gene expression behavior.

in the compendium, centering on the biological processes and/or genes they are interested in. In that sense, it is very much an exploratory tool, meant to assist more dedicated research in grapevine genomics, biology, and physiology, even if the integration of over 1500 transcriptomics samples into a single data set can be quite powerful in and of itself. The case studies presented in the results are examples of the type of analyses that can be done with VESPUCCI, and the sort of insights that can be gained from the combined data in the compendium. They all represent cases where VESPUCCI shows interesting modular gene expression responses that were not known previously, whether from the individual experiments included in the compendium, from published papers, or from other, independent (even non-transcriptomics) experiments or sources of information. In contrast to model organisms for which available -omics experiments are considerable, crop species usually lack of a substantial amount of data. Nevertheless, there is an increasing interest for a more systemic view of crop species [68], [69], driven by the ever-decreasing cost of high-throughput technologies and the development of new analysis tools. The availability of transcriptomics technology has increased substantially during recent years. Nowadays, RNA-Seq experiments enable scientists to reliably measure the majority of expressed genes. However, during the early days of transcriptomics, microarray measurements often comprised only a part of the complete transcriptome. The end result is that across the entire VESPUCCI gene expression compendium, the proportion of missing values is substantial (36%). Even though the great majority of samples have been measured using the Nimblegen or RNA-Seq technologies which can both cover the near complete transcriptome, the probes of the other microarray platforms are not able to provide measurements for as many genes. This is irremediable and intrinsic to the source data. We dealt with it by attempting to provide optimal, as reliable as possible expression measurements across the compendium, both at the level of the actual probe-to-gene mapping, as well as at the level of defining of the list of measurable gene transcripts. These measurable transcripts (representing the rows of the gene expression ma-

trix) incur some limitations in and of themselves as they are entirely based on the gene predictions for *V. vinifera* cv. Pinot Noir, with implications for experiments done on other cultivars. When microarray experiments are performed to measure expression for a specific cultivar with platforms containing probes designed from different cultivars, this generally leads to poorer signals, given the impossibility to distinguish expression variability from genomic differences among those cultivars. The reason is the lack of available high-quality gene predictions for each cultivar. While RNA-Seq has the advantage of enhancing its value over time with better genomes and gene annotations by re-doing the transcriptome mapping on the appropriate cultivar, the situation is more complicated for microarray data. The solution is never ideal as nothing can be done to increase the quality of intensity signals if there is a mismatch between the cultivar used to design the probes, and the one used to do the experiment. Nevertheless, remapping the microarray probes on the cultivar-specific genes of the experiment would improve the gene annotation of the array platform and ensure only the reliable probes are considered to generate the final expression values. A novelty in the latest release of COLOMBOS is the option to explicitly recognize genomic differences between strains or cultivars instead of using a single reference genome to represent a species. This improves read alignment (RNA-Seq) or probe-to-gene mapping (microarrays) and generates higher quality expression data. In the long term, as more grapevine cultivar genomes become available, we can rely on these COLOMBOS innovations to build compendia for different cultivars and integrate them at the species level using homolog mappings, creating a proper 'meta-compendium' for grapevine varieties. Currently VESPUCCI is limited to our knowledge of the *V. vinifera* cv. Pinot Noir genome, and despite the existence of a more recent version of the CRIBI gene prediction [70], we decided to keep V1 as the basis for this first release. From a practical perspective, by the time V2 was made publicly available, most of the compendium was already built and the switch to the newer version didn't show a significant increase quality-wise. The great majority of genes does not change in terms of gene structure, and as such for our purposes the end result was largely unaffected by the enhancements of the newer version over V1. Nevertheless, as the number of experiments (especially RNA-Seq) increases, the benefits of relying on V2 will become more prominent; for future VESPUCCI releases, we will most likely shift toward V2 (or more recent versions) to take advantage of the extended UTR regions for which NGS technologies provide better measurements. The measurable gene transcripts that, we defined do not correspond one-to-one to the CRIBI gene predictions, but instead contain some 'gene clusters'. Expression data for these gene clusters are a compromise between our ability to measure each and every single gene individually, and how many genes can be reliably measured in total. While not absolute proof that these probes are unable to adequately distinguish the intended target genes, our results (Figures 4.1 and 4.2, and Supplementary Materials) showed that it is almost impossible to measure differences between each single gene in the clus-

ters. This supports our decision to throw them together: even if these probes were capable of capturing different transcripts, the results do not indicate that this was the case for the more than 500 Nimblegen sample contrasts in the compendium. Therefore, instead of discarding the shared probes and lose potentially valuable information, we accepted the impossibility to unambiguously discriminate each and every single gene, gaining the opportunity to have a single measurement for those gene transcripts as a whole. Note that while the issue itself is (microarray) platform specific, the proposed 'gene clusters' are not. We chose to define them based on the platform with the highest data representation: the Nimblegen platform holds the largest number of samples as well as the highest quality of probe-to-gene mapping. This has no detrimental effect on data from the other microarray platforms, but RNA-Seq technology can provide individual gene measurements for at least some of our defined clusters (given that the corresponding gene sequences show enough dissimilarity). Due to the current low number of RNA-Seq experiments compared to the Nimblegen ones, we decided on clustering genes together in measurable sets to get the best out of all the data as a whole. As soon as RNA-Seq experiments will be more prevalent, we will revise the gene clusters to gain the ability to measure more genes separately for RNA-Seq, at the expense of losing the corresponding probes on the Nimblegen platform. VESPUCCI includes nearly all of the gene expression data that is publicly available for grapevine at the moment; it provides a snapshot of the current situation of transcriptomics experiments performed. We're planning to keep it up to date by releasing yearly content updates. In the current release, berry development studies are the most represented experiments (especially during véraison) and this comes with no surprise given the importance of fruit quality in wine and spirits' production. This will be all the more obvious when mining for genes related to fruit ripening. Given the complexity of this developmental process, in which the fruit undergoes radical phenotypic and biochemical modifications (related to shape, size, color, sugar and aroma content, etc.), the number of modulated genes is quite big. VESPUCCI is meant as an exploratory tool to help researcher not only in finding patterns of gene expression for genes of interest, but also to aid the design of new experiments providing the most complete transcriptomics information currently available.

Part two

CHAPTER 5

Modelling changes in gene expression

The ability to model changes in gene expression is a valuable characteristic in nowadays analysis as it enables a deeper understanding of the biology of the system under investigation and the ability to predict possible behavior as response to changes in biological conditions. Sample size greatly affect our ability to generalize results and an adequate magnitude of data is always a desirable feature. In previous chapters we described how transcriptomic data could be integrated into a massive, coherent gene expression matrix. In the next chapters we will discuss how to draw inference from such data to facilitate biological knowledge discovery. We adopted two mathematical models: a statistical framework that provides a *pivotal* tool for all subsequent analysis, used to identify gene expression changes responsible for the observed values, and a model that exploits it together with prior structural knowledge of genetic sensory-response mechanisms in order to model and, possibly, predict changes in gene expression induced by shifting biological conditions. Such approaches have several advantages that are relevant given the particular nature of data we want to model. The former model employs a Bayesian approach, in which we model a probability distribution over an underlying change in expression for a given gene in response to a given condition. The inherent sequential nature of Bayesian learning makes it well-suited for the disparateness in the number of replicates in the expression compendium. Moreover, the Bayesian approach provides a convenient way for introducing prior knowledge given by properties of the data and its pre-processing and finally it models a complete posterior distribution instead of point estimators. The latter approach uses Boolean networks to model structural information about the (partially) known genetic mechanisms of response to stimuli, representing such mechanisms on the level of a single cell. Assuming that gene expression measurements have been taken from a population of cells

in different steady-state conditions, we fit a distribution of attractor states to the posterior over the underlying change in expression as determined from the observed measurements by the Bayesian model.

5.1 A Bayesian approach

5.1.1 Bayesian statistics

Statistical inference deals with the estimation of unknown parameters describing some population properties via the observation of some data and the use of statistical models that links data to parameters ([71]). While in frequentist statistics parameters are fixed values, Bayesian statistics consider parameters itself as being a random variable with an assigned probability distribution, known as *a prior* distribution. Therefore, Bayesian statistics define probability as a means to quantify uncertainty (degree of subjective belief) about the value of unknown parameters. The peculiarity of this approach lies in the possibility to *update* the prior belief as soon as new data are available to yield a *posterior* belief (after having observed the data) to express what is known about parameters given both the sample data and the *prior* information. If we represent the data by the symbol D and the set of unknown parameters by θ, then we can specify a joint probability distribution over data and parameters $p(D, \theta)$. By the definition of conditional probability:

$$p(D, \theta) = p(D|\theta) \cdot p(\theta)$$

The term $p(D|\theta)$ is the *likelihood* and embodies the statistical model while the term $p(\theta)$ is the *prior* distribution and quantifies the belief (before observing any data) about parameters. With the application of the Bayes's theorem we synthesise these two sources of information through the equation:

$$\underbrace{p(\theta|D)}_{posterior} = \frac{\overbrace{p(D|\theta)}^{likelihood} \times \overbrace{p(\theta)}^{prior}}{\underbrace{p(D)}_{marginal}}$$

The term $p(D)$ is called the *marginal distribution* and acts as normalizing factor, while $p(\theta|D)$ is called the *posterior* distribution for θ (given the data) and expresses what is known about θ based on both the sample data and the *prior* information. Posterior is proportional to prior times likelihood, and thus it can also be rewritten has:

$$\underbrace{p(\theta|D)}_{posterior} \propto \underbrace{p(D|\theta)}_{likelihood} \times \underbrace{p(\theta)}_{prior}$$

5.1.2 A Bayesian noise model

Figure 5.1 a) is a schematic view of the compendium and how it determines the parameter structure for prior, likelihood and posterior distributions. The green column is one condition contrast (composed by n replicated sample contrasts) and represent the knowledge that would render the prior distribution in the model, i.e. the probability over the true underlying change in gene expression (μ_x) given that we have not observed any data yet (X). This information is related to the pre-processing and subsequent quality control of the data. From this we know that the distribution of log-ratios for a single column of the compendium matrix (i.e. a sample contrast) is symmetrical around 0 and has a minor fraction of 'outliers' that fall well above or below it, exhibiting a strong over or under expression respectively. We model the prior distribution as a mixture of 3 Gaussian distributions, one for the bulk of genes around 0, one for strong overexpressed genes and one for strong underexpressed genes. On the other hand, the specific information for a single gene, the orange row in figure 5.1 a), that we would include in the model is related to the notion that some genes may be inherently more variable than others when measured multiple times across conditions that can be considered biological replicates. The precision γ (precision here denotes the reciprocal of variance) represents the inherent gene variability as a Gaussian noise with mean 0. The distribution parameters are fitted from the data in the compendium using an interative algorithm that maximize the evidence function similarly to an Expectation–Maximization (EM) algorithm. Figure 5.1 b) and c) show how the posterior distribution would get updated as new data (replicated measurements) are observed.

The posterior probability distribution, in blue in figure 5.1 a), represent the probability over the underlying true change in gene expression μ_x for a given gene G in response to a given condition contrast C, with n replicated measurement $X = (x_1, \ldots, x_n)^T$. Modelling it as a posterior probability distribution, and thus following Bayes rule gives:

$$p(\mu_x|X, G, C) = p(X|\mu_x, G, C) \cdot p(\mu_x, G, C) \tag{5.1}$$

If we assume the inherent gene variability γ to be independent from the condition contrast in which gene have been measured, then G is irrelevant for the prior distribution of μ_x and similarly C is irrelevant for the likelihood of X, so we can simplify the posterior distribution as:

$$p(\mu_x|X, G, C) = p(X|\mu_x, G) \cdot p(\mu_x, C)$$

Since the prior distribution is a mixture of $k = 3$ Gaussian distributions so will be the posterior. We can rewrite the above posterior as:

$$p(\mu_x|X, M, A, \Pi, \gamma) = p(X|\mu_x, \gamma) \cdot p(\mu_x, M, A, \Pi) \cdot c^{-1}$$

With $M = (\mu_1, \ldots, \mu_k)$, $A = (\alpha_1, \ldots, \alpha_k)$, $\Pi = (\pi_1, \ldots, \pi_k)$ being means, precision and mixture weights respectively of the $k = 3$ components in the

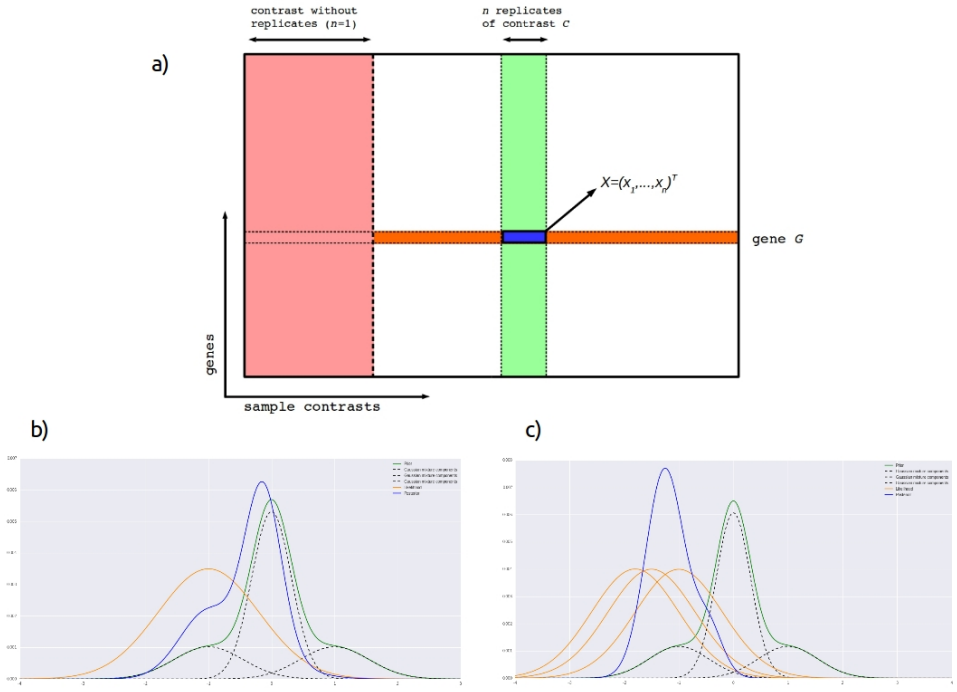

Figure 5.1: **Bayesian modelling overview** The whole compendium **a)** is a matrix in which rows represent single genes and columns condition contrasts. The posterior distribution over the true underlying change in gene expression for a single gene in a single condition contrast (in blue) gets updated (from **b)** to **c)**) as new replicated measurements are observed (likelihood distribution in orange). Prior distribution (in green) represent the probability over the underlying change in gene expression for a single condition contrast before observing any data. It is modelled as a mixture of 3 Gaussian distributions (dashed black lines in **b)** and **c)**), one for overexpressed genes, one for underexpressed genes and the last one for gene that do not have a notable change in expression.

mixture model. c here is a normalization factor to ensure that the posterior integrates to 1.

5.1.3 Analytical form of the posterior

Since the prior distribution is a mixture of k Normal distributions, the posterior will also be a mixture of k Normal distributions:

$$\begin{aligned} p(\mu_x|X) &= p(X|\mu_x)p(\mu_x)c^{-1} \\ &= c^{-1}p(X|\mu_x)\sum_{i=1}^{k}\pi_i p(\mu_x|i) \\ &= c^{-1}\sum_{i=1}^{k}\pi_i p(\mu_x|\mu_i,\alpha_i)p(X|\mu_x) \end{aligned} \tag{5.2}$$

where

$$p(\mu_x|\mu_i,\alpha_i) = \mathcal{N}(\mu_i,\alpha_i^{-1}) \tag{5.3}$$

and

$$\begin{aligned} p(X|\mu_x) &= \prod_{j=1}^{n} p(x_j|\mu_x,\gamma) \\ &= \prod_{j=1}^{n} \mathcal{N}(x_j-\mu_x|0,2\gamma^{-1}) \\ &= \prod_{j=1}^{n} \mathcal{N}(x_j,2\gamma^{-1}) \end{aligned} \tag{5.4}$$

The variance of the likelihood factors is arbitrarily set to $2\gamma^{-1}$, since the assumption is that γ manifests itself on the log expression values of individuals samples so that their differences have twice the variance. We can then rewrite 5.2 as:

$$p(\mu_x|X) = c^{-1}\sum_{i=1}^{k}\pi_i w_i \mathcal{N}(\Theta_i,\beta_i^{-1}) \tag{5.5}$$

with

$$\Theta_i = \frac{\alpha_i \mu_i + \frac{\gamma}{2} \sum_{j=1}^{n} x_j}{\alpha_i + \frac{n\gamma}{2}}$$

$$\beta_i = \alpha_i + \frac{n\gamma}{2}$$

$$w_i = \mathcal{N}\left(\mu_i \middle| \frac{\sum_{j=1}^{n} x_j}{n}, \alpha_i^{-1} + 2(n\gamma)^{-1} \right) \prod_{j=2}^{n} \mathcal{N}\left(x_j \middle| \frac{\sum_{m=1}^{j-1} x_m}{j-1}, \left(\frac{j}{j-1} \right) 2\gamma^{-1} \right)$$

$$c = \sum_{i=1}^{k} \pi_i w_i$$

The proof of 5.5 is given in A.3

5.1.4 Fitting unknown parameters

In a standard Bayesian framework, the prior distribution is fixed before any data are observed and thus the prior parameters $M = (\mu_1, \ldots, \mu_k)$, $A = (\alpha_1, \ldots, \alpha_k)$, $\Pi = (\pi_1, \ldots, \pi_k)$ are known. Since M, A and Π are unknown we can use a methods called empirical Bayes (also known as evidence approximation) to obtain an estimation of the prior parameters from the data. The initial estimation of parameters M, A and Π are obtained, for each contrast, fitting a mixture of $k = 3$ Gaussian distribution using an Expectation-Maximization (EM) iterative schema, with some constraints on the parameters. That is, the middle Gaussian distribution has mean equal to 0 since we know that the distribution of logratios for a single contrast is for the largest part symmetrical around 0 and can be approximated by a robustly fit normal distribution. Moreover, the means for the other 2 Gaussian distributions are set to be at least one standard deviation (of the middle Gaussian) away from 0 and are fitted to the portion of genes that show a strong under and over-expression. The precision γ, that represents the inherent gene variability, is estimated initially from the mean of the unbiased standard deviations of the replicated measurements of each single gene. In the empirical Bayes framework the values of M, A, Π and γ are obtained by maximizing the marginal likelihood function $p(X|M, A, \Pi, \gamma)$. The marginal likelihood function $p(X|M, A, \Pi, \gamma)$ is obtained by integrating over the parameter μ_x, so that:

$$\begin{aligned} p(X|M, A, \Pi, \gamma) &= \int p(X, \mu_x|M, A, \Pi, \gamma) d\mu_x \\ &= \int p(X|\mu_x, M, A, \Pi, \gamma) p(\mu_x|M, A, \Pi, \gamma) d\mu_x \end{aligned}$$

Since (as we already stated in 5.1) we assume that γ is independent from the condition contrast in which gene have been measured, then we can simplify the marginal likelihood as:

$$p(X|M, A, \Pi, \gamma) = \int p(X|\mu_x, \gamma)p(\mu_x|M, A, \Pi)d\mu_x$$

where $p(X|\mu_x, \gamma)$ is distributed as the likelihood in our model 5.4 and $p(\mu_x|M, A, \Pi)$ has a distribution like the prior in our model 5.3, and thus:

$$\begin{aligned} p(X|M, A, \Pi, \gamma) &= \int \prod_{j=1}^{n} \mathcal{N}(x_j|\mu_x, 2\gamma^{-1}) \sum_{i=1}^{k} \pi_i \mathcal{N}(\mu_x|\mu_i, \alpha_i^{-1})d\mu_x \\ &= \int \sum_{i=1}^{k} \pi_i w_i \mathcal{N}(\mu_x|\Theta_i, \beta_i^{-1})d\mu_x \end{aligned}$$

with Θ_i, β_i and w_i as in 5.5. Since w_i is independent from μ_x, we can reformulate the integral as:

$$= \sum_{i=1}^{k} \pi_i w_i \int \mathcal{N}(\mu_x|\Theta_i, \beta_i^{-1})d\mu_x$$

By definition the integral:

$$\int \mathcal{N}(\mu_x|\Theta_i, \beta_i^{-1})d\mu_x = 1$$

and thus we are left with:

$$p(X|M, A, \Pi, \gamma) = \sum_{i=1}^{k} \pi_i w_i \tag{5.6}$$

where π_i and w_i are the same as in 5.5. In order to correctly estimate parameters $M = (\mu_1, \ldots, \mu_k)$, $A = (\alpha_1, \ldots, \alpha_k)$, $\Pi = (\pi_1, \ldots, \pi_k)$ and γ, though, 5.6 should be rewritten taking in consideration *all* measurements across the entire compendium. That is:

$$\begin{aligned} p(X|M, A, \Pi, \gamma) = p(X_{1,1}|M_1, A_1, \Pi_1, \gamma_1) \times \ldots \times p(X_{i,j}|M_j, A_j, \Pi_j, \gamma_i) \times \ldots \\ \times p(X_{n,m}|M_m, A_m, \Pi_m, \gamma_n) \end{aligned} \tag{5.7}$$

where m is the total number of contrasts and n is the total number of genes in a compendium. The reason for the need to estimate the parameters across all data is given by the dependency structure of the model, i.e. each γ is calculated starting from *all* contrasts, and thus *depends* on all M, A and Π. Similarly, prior parameters are estimated based on *all* genes, and thus *depends* on all γ. In order to maximize 5.7 with respect to $M = (\mu_1, \ldots, \mu_k)$, $A = (\alpha_1, \ldots, \alpha_k)$,

$\Pi = (\pi_1, \ldots, \pi_k)$ and γ we used an EM interative schema in which parameters estimation is refined on every iteration using previous iteration results. Assuming each gene to be independent from each other, new γ are estimated based on prior parameters M, A and Π. Similarly, prior parameters are estimated based on previous γ values. Therefore, to maximize M, A, Π and γ we defined:

$$f(\gamma) = \prod_{h=1}^{m} p(X_h|M_h, A_h, \Pi_h, \gamma)$$
$$g(M, A, \Pi) = \prod_{v=1}^{n} p(X|M, A, \Pi, \gamma_v)$$

Where m is the number of contrasts and n is the number of gene in the compendium. In practice it is more convenient to work with the natural logarithm, and thus:

$$\begin{aligned} \ln f(\gamma) &= \ln \prod_{h=1}^{m} p(X_h|M_h, A_h, \Pi_h, \gamma) \\ &= \sum_{h=1}^{m} \ln p(X_h|M_h, A_h, \Pi_h, \gamma) \end{aligned} \tag{5.8}$$

and

$$\begin{aligned} \ln g(M, A, \Pi) &= \ln \prod_{v=1}^{n} p(X|M, A, \Pi, \gamma_v) \\ &= \sum_{v=1}^{n} \ln p(X|M, A, \Pi, \gamma_v) \end{aligned} \tag{5.9}$$

5.1.5 Implementation

In order to find the value of γ that maximize 5.8 and the values for M, A and Π that maximize 5.9, we implemented an EM iterative schema in Python using a non-linear optimization package [72]. Since we assume the γ parameters for each gene are independent from each other and the same goes for the prior distribution parameters for the contrasts, the calculations are performed in parallel for each row (gene) and column (contrast). The termination of the algorithm is controlled by a limit on the number of iterations and a condition on the improvement with respect to previous iteration that has to be bigger than a given cut-off in order to proceed with the next iteration.

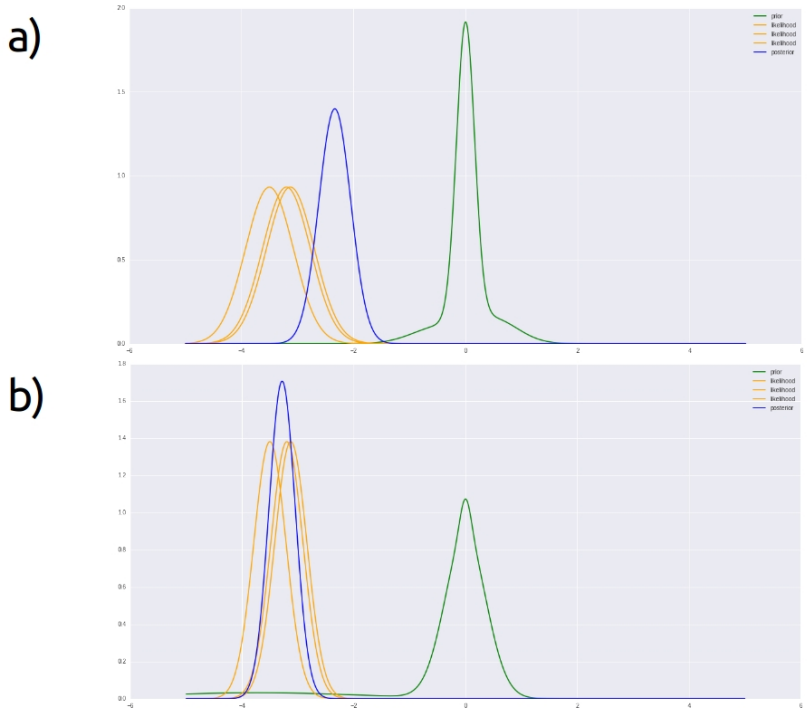

Figure 5.2: **Noise model parameters fit a)** the initial fit of parameters M, A, Π and γ for a gene-contrast combination. Prior and posterior distributions are in green and blue respectively while likelihood are drawn in orange. In **b)** the same gene-contrast combination after the parameters estimation with the Empirical Bayes approach.

5.2 A Boolean approach

5.2.1 Boolean networks

Boolean networks[73] are sets of Boolean variables (i.e. variables having just two values, usually denoted *true* and *false*), and Boolean function (i.e. a function with the form $f : B^k \to B$, where $B = true, false$ are connected by logical operators and, or, not and $k \in \mathbb{N}$). Each variable (a node in the network) has associated a function, with inputs the states of the nodes connected to it. Boolean networks have a fixed topology that doesn't change with time, but they are dynamic in the sense that states of nodes (values of Boolean variables) change synchronously and discreetly with time. If the state of a node v_i at time t is denoted as $x_i(t)$, then the state of that same node at time $t + 1$ is given by: $x_i(t + 1) = b_i(x_{i1}, \ldots, x_{ik})$ where x_{ij} are the states of the nodes connected to v_i and b_i is a Boolean function. Since a Boolean network has only 2^n possible states (with n being the number of nodes) and their dynamics are deterministic a trajectory will, sooner or later, reach a previously visited state. Once this will happen once, the trajectory will be stuck to repeatedly visit the same nodes forever. These cycles of states into which the Boolean networks 'falls' are called *attractor* states. Attractors are particularly interesting as they capture the intuition of steady-states for living systems, i.e. a situation in which all state variables are constant in spite of ongoing processes that strive to change them.

5.2.2 The model

Every measurement in the compendia is a difference in expression (in logarithmic scale) between two conditions. Since we assume that:

1. every sample is composed by a *population* of cells;
2. every sample have been measured during cells *steady-state*;

the actual measurements in the compendia for a single gene in a given condition contrast do not represent one single 'behavior' coming from that gene in a single cell, but (more precisely) represent an 'average behavior' for that gene measured in the entire population of cells the sample is composed of. Even though expression data in the COLOMBOS compendia are not well suited for being used as-is with Boolean networks (since they usually don't represent time-series and so it's impossible to model the time-dependency of expression changes), they still provide both a simple model for describing interactions of how a single cell responds to stimuli, and a convenient way to describe each possible steady-state in which a system (a living cell in our case) can be at any given time, through the concept of attractor states. Since we cannot assume all cells in the sample to be in the same steady-state, we are left with the problem of estimating how many cells (in proportion) are in a specific steady-state in order to explain the

expression, or better the 'mixture' of expressions, we observe. Thus, given a shift in biological conditions, expression data in the compendia for those same genes (known to be involved) are used to fit a model that uses Boolean network attractor states to *simulate* the different steady-states in which sub-populations of cells were during the time of measurements. Estimated weights for each possible attractor state represent then the proportion of cells in that particular steady-state. Since we don't directly fit, but rather simulate, a Boolean network, prior knowledge on its topology is required in order to take advantage of compendia gene expression data. Boolean networks are manually built starting from the concept of 'genetic sensory response units' (Gensor Units) [75]. Gensor Units (see figure 5.3) are relatively small modules that encapsulate the concept of a signal sensed by the cell, the transduction of the signal and the transcriptional regulation that modifies expression of genes responsible for the response. Gensor Units have several advantages since they are:

1. highly characterized;
2. manually curated;
3. modular;
4. relatively small;
5. heterogeneous.

Their specificity for a single condition input signal, highly characterization and modularity provide an ideal framework for the methodology in degrees of increasing complexity. However, even if Boolean network simplicity permits to work with relatively large networks, the complexity of the problem, once several of such units are combined, is combinatorial. Thus initially we limited simulated condition contrasts to be just four per GU in order to keep the computation tractable: the combination of the activation (or inactivation) of signal and transcription factor.

5.2.3 Implementation

The complete procedure as it is currently implemented starts from one (or more) given Gensor Units converted into a Boolean network, then it:

1. **simulates test and reference conditions**: setting nodes corresponding to signal and transcription factor to all the 4 combinations of *true* and *false* (to simulate the presence/absence) in the Boolean network;
2. **calculate all possible attractor states for the two networks**: to define Boolean networks and calculate all possible attractor state we used the R package BoolNet [76];

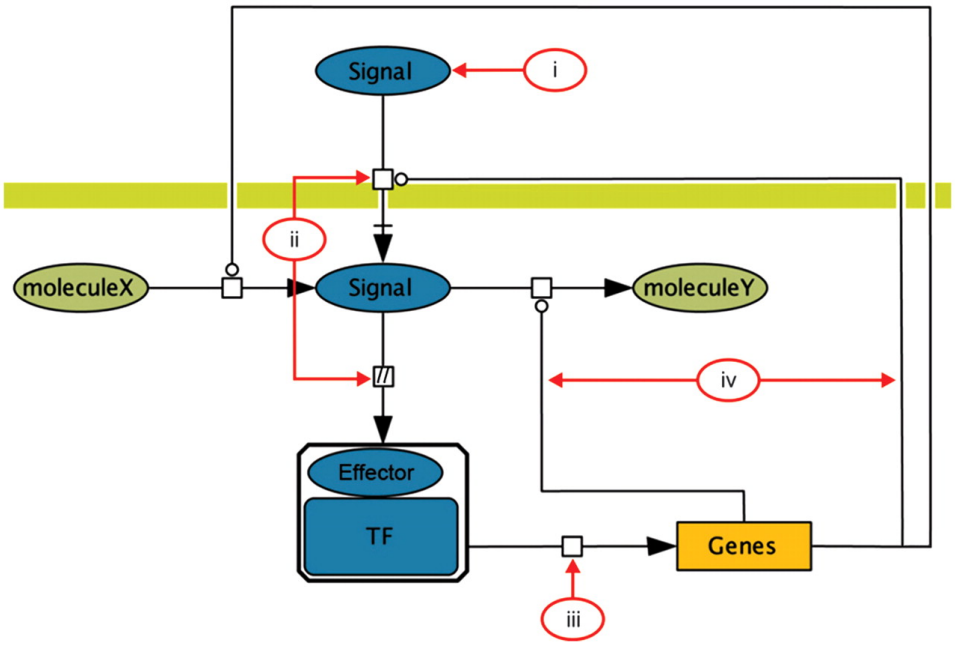

Figure 5.3: **Gensor Unit** A Gensor Unit is composed by an extra-cellular signaling molecule that activates a specific receptor that triggers a biochemical chain of events inside the cell involving the activation of a transcription factor that modifies the expression of genes responsible for the response and that control the signal in a positive (or negative) feedback loop.

3. **calculate expression level for each network**: as the relative value of expressed genes over all attractor states. That is, the ratio between the number of time a gene node value is *true* over the total number of attractor states;

4. **calculate the simulated contrasts expression values**: as the difference between test and reference expression values;

5. **find the best weight for the attractor states**: currently implemented as an optimization problem:

$$\max_{\mathbf{x},\mathbf{y}\in\mathbb{R}^n,\mathbf{w}\in\mathbb{R}^m} f_{\mathbf{w}}(\mathbf{x},\mathbf{y}) \tag{5.10}$$

with

$$\sum_{i=1}^{m} w_i = 1$$

where f is the cosine similarity function, **x** is a simulated contrast calculated in the previous step, **y** is a real contrast [1] in the compendia with the same *test* and *reference* conditions, **w** is the vector of weights for the attractor states, n is the number of genes and m is the number of attractors.

The need to fit the weights to real contrasts is justified by the fact that otherwise all attractor states would be considered equally important. Namely, all subpopulation of cells in different steady-state in the measured sample would be equally numerous. Once the model is fit, it would be possible, given an unknown contrast to understand which signal (if any) has been sensed by the cells in the sample. The real benefit, however, would rise combining more Gensor Units at the same time to model signal interactions.

[1] Actually it is a vector of point-estimator like maximum a posteriori (MAP) from the posterior over the underlying true change in gene expression as determined from the observed measurements by the Bayesian model.

CHAPTER 6

Conclusion, discussion and future perspectives

The technological possibilities that arose during last few decades have give modern biologists a lot of opportunities in terms of system-wide experiments measuring a lot of features at the same time. Unfortunately, this comes with the production of a considerable amount of data for which at least basic skills on data-management are required. The charm of new high-throughput technologies urges scientists to adopt them, sometimes without having a clear idea about limits and possibilities, or the new sources of noise that accompany them and of which little may be known. Moreover, the sole adoption of new technologies tends to be considered adequate by scientific journals to justify a publication. This tendency has led to the situation in which data production is often considered the goal of a scientific effort instead of being considered just a tool towards answering a scientific question. Since data alone aren't sufficient to create knowledge, the adoption of statistical methods is fundamental to extract valuable information and attempt to give valid biological interpretations of data. In this context data integration is a valuable approach since:

- it ease the issues about data and noise management and allow a broad view on existing experiments;
- and an adequate magnitude of data is a necessary characteristic to be able to draw strong and valid conclusions.

This Ph.D. thesis tries to tackles both issues showing the implementation and the application of a methodology for gene expression data integration, and the development of statistical tools to analyze them.

The first part of the thesis focused on the work that was done related to the COLOMBOS gene expression compendia technology. The integration of transcriptomic, and in general different -omics data, is an inherent part of biology

nowadays. The shift towards a data-driven science started with the advent of high-throughput technologies that are able to produce an impressive amount of data measuring several layers of information at a broad level. Currently the quantity of data produced outweighs our ability to analyze them [77],[78] and often not all the information a dataset can provide is used (for example RNA-seq experiment used only for quantification, ignoring the actual sequence information, or sequencing the whole genome to extract only mitochondria information for phylogeny reconstruction). Bioinformatics and computational biology are essential tools for life-sciences because of it, and is testified to by the number of publications that use bioinformatics tools and databases [79]. In this context of need for data integration, the COLOMBOS approach to transcriptomic integration proved to be useful [23],[24],[25]. The entire expression compendia have been used as well, for transcriptional regulatory network inference [26], for the creation of co-expression networks [27],[28], and to study the evolution of regulatory networks [21]. They also served to study Mutation Rate Plasticity (MRP) allowing a fast analysis of all published studies in gene expression [20], or provided essential data for a multi-omics data integration using network analysis and flux balance analysis [22]. The COLOMBOS controlled vocabulary used to formally annotate condition contrasts has provided a way to link changes in gene expression to causal factors like genomic mutations [29], or the activation of transcription regulation [30].

The maintenance and development of COLOMBOS (and COMMAND, the compendia build tools) has been an ongoing process and in chapter 2 we discussed how we initialized a radical change. The stepwise phase-out strategy adopted here to gradually disband the old implementation, has permitted to use the newer version as soon as it became a working prototype while preserving the old implementation for all the features not yet implemented in the newer version. The development of COLOMBOS old code will stop and the whole application will be moved to the new framework, implementing a new paradigm for meta-compendia allowing a more natural and powerful implementation of cross-species analysis that is now bound to the limit of the current version. Moreover, the noise model (described in section 5.1), or a future version thereof, will be included to provide a statistical layer on which several statistical tools would be developed (like identification of differentially expressed genes, mining complex patterns of co-expression and the use of the Boolean network framework described in section 5.2).

COLOMBOS was originally conceived for three prokaryotes model organisms, *Escherichia coli*, *Bacillus subtilis* and *Salmonella enterica serovar Typhimurium* and got extended over the years including several non-model bacterial species. The latest version of COLOMBOS has been applied also to the plant crop *Vitis vinifera* [31] creating a comprehensive tool for gene expression data exploration for the grapevine community. The issue of data integration for non-model organisms is even more important given the lesser amount of available data, that instead would provide more insight when gathered together

[64]. Alongside the creation of a grapevine gene expression compendium, we started the creation of a compendium for apple (*Malus x domestica*). In this case, however, there was little to no publicly available gene expression data and a reliable list of genes was lacking. Thus, our effort has been devoted more to the creation of such a list of genes to be used as starting point for measuring gene expression. An iterative process of transcriptome sequence assembly that starts from short and reliable sequences from various sources of data (Roche 454 and Illumina sequences) was developed to refine and elongate transcript sequences by exploiting the genome sequence to correctly group together sequence reads potentially coming from the same transcript. Although the final list has already been used in different projects, the number of gene expression experiments from public databases, as well as those produced inside the Fondazione Edmund Mach, does not yet justify the creation of a gene expression compendium and so this work was not included as part of this thesis.

COLOMBOS primary objective is first and foremost to bring together as much data as possible allowing the exploration and research for complex patterns of (co)expression. The second part of this thesis was focused on the sort of biological knowledge we could gain from such expansive expression data sets. One of the disadvantage of its current implementations however, is the impossibility to perform rigorous statistical inference, because of the peculiar nature of the data, like the varying number of replicates, the existence of contrasts that measure only biological variability and the dependency of reference samples. In chapter 5 we developed a statistical framework with the purpose to overcome some of such limitations. The Bayesian noise model developed in section 5.1 serves as a basis for 'interrogating' the data in a statistically sound way. Notwithstanding the possibility to perform statistical analysis, there is still room for improvements together with issues to be addressed. The likelihood in 5.4 is a particular case and represents the most straightforward situation where all the logratios in X are independent. This is only partially true. In reality a lot of measurements for a condition contrast are dependent by sharing a reference sample. If that reference sample is also replicated, this will generally result in an additional *self-self* contrast which measures at most biological variability. One possible solution would be to reformulate the original data in the compendia in order to make them represent logratios of single test samples against the average of the reference samples, and thus removing *self-self* contrasts. Expression logaratios X of this type (defined against the same average reference sample) are still dependent, and as such suffer from an unknown bias δ that then should be incorporated in the likelihood expression to be marginalized out and representing the difference between the average reference sample expression over r replicates and the true underlying expression of that reference condition. The implementation of the EM schema to fit M, A, Π and γ can be improved as well by using gradients to help the search algorithm to converge faster, since at the moment it uses a derivative-free method.

While the Bayesian noise model provides a basic statistical framework to deal

with the inherent variability of genes, the Boolean network approach (described in section 5.2) exploit the prior knowledge about relationships among genes (as given by Gensor Units) to provide new information about the possible stimuli and conditions that lead to observed gene expression measurements. Despite the potential of such approach, the benefits are bound to be untapped since the lack of appropriate data to be used to fit the model.

Appendices

APPENDIX A

Proofs

Lemma 1.

$$\mathcal{N}(\mu_1, \alpha_1^{-1})\mathcal{N}(\mu_2, \alpha_2^{-1}) = c\mathcal{N}(\mu, \alpha^{-1}) \tag{A.1}$$

with

$$\begin{aligned}\mu &= \frac{\alpha_1\mu_1 + \alpha_2\mu_2}{\alpha_1 + \alpha_2}\\ \alpha &= \alpha_1 + \alpha_2\\ c &= \mathcal{N}(\mu_1|\mu_2, \alpha_1^{-1} + \alpha_2^{-1})\end{aligned}$$

Proof. Let $f(x) = \mathcal{N}(x|\mu_1, \alpha_1^{-1})$ and $g(x) = \mathcal{N}(x|\mu_2, \alpha_2^{-1})$

$$f(x)g(x) = \frac{\sqrt{\alpha_1\alpha_2}}{2\pi}e^{-\frac{1}{2}\underbrace{((x-\mu_1)^2\alpha_1 + (x-\mu_2)^2\alpha_2)}_{\beta}}$$

Let $\beta = (x-\mu_1)^2\alpha_1 + (x-\mu_2)^2\alpha_2)$

$$\begin{aligned}\beta &= (x^2 - 2x\mu_1 + \mu_1^2)\alpha_1 + (x^2 - 2x\mu_2 + \mu_2^2)\alpha_2\\ &= x^2\alpha_1 - 2x\mu_1\alpha_1 + \mu_1^2\alpha_1 + x^2\alpha_2 - 2x\mu_2\alpha_2 + \mu_2^2\alpha_2\\ &= x^2(\alpha_1 + \alpha_2) - 2x(\mu_1\alpha_1 + \mu_2\alpha_2) + (\mu_1^2\alpha_1 + \mu_2^2\alpha_2)\\ \frac{\beta}{(\alpha_1 + \alpha_2)} &= x^2 - 2x\left(\frac{\mu_1\alpha_1 + \mu_2\alpha_2}{\alpha_1 + \alpha_2}\right) + \left(\frac{\mu_1^2\alpha_1 + \mu_2^2\alpha_2}{\alpha_1 + \alpha_2}\right)\\ \frac{\beta}{(\alpha_1 + \alpha_2)} &= x^2 - 2x\left(\frac{\mu_1\alpha_1 + \mu_2\alpha_2}{\alpha_1 + \alpha_2}\right) + \left(\frac{\mu_1^2\alpha_1 + \mu_2^2\alpha_2}{\alpha_1 + \alpha_2}\right) + \left(\frac{\mu_1\alpha_1 + \mu_2\alpha_2}{\alpha_1 + \alpha_2}\right)^2 - \left(\frac{\mu_1\alpha_1 + \mu_2\alpha_2}{\alpha_1 + \alpha_2}\right)^2\end{aligned}$$

Let $\mu = \frac{\mu_1\alpha_1+\mu_2\alpha_2}{\alpha_1+\alpha_2}$ and $\alpha = \alpha_1 + \alpha_2$

$$\begin{aligned}
\beta &= (x-\mu)^2\alpha + (\mu_1-\mu_2)^2\left(\frac{\alpha_1\alpha_2}{\alpha_1+\alpha_2}\right) \\
&= \frac{(x-\mu)^2}{\alpha^{-1}} + \frac{(\mu_1-\mu_2)^2}{\alpha_1^{-1}+\alpha_2^{-1}} \\
f(x)g(x) &= \frac{\sqrt{\alpha_1\alpha_2}}{2\pi} e^{-\frac{1}{2}\left(\frac{(x-\mu)^2}{\alpha^{-1}}+\frac{(\mu_1-\mu_2)^2}{\alpha_1^{-1}+\alpha_2^{-1}}\right)} \\
&= \frac{\sqrt{\alpha\frac{1}{\alpha_1^{-1}+\alpha_2^{-1}}}}{2\pi} e^{-\frac{1}{2}\left(\frac{(x-\mu)^2}{\alpha^{-1}}+\frac{(\mu_1-\mu_2)^2}{\alpha_1^{-1}+\alpha_2^{-1}}\right)} \\
&= \sqrt{\frac{\alpha}{2\pi}} e^{-\frac{1}{2}(x-\mu)^2\alpha} \sqrt{\frac{1}{2\pi(\alpha_1^{-1}+\alpha_2^{-1})}} e^{-\frac{1}{2}\left(\frac{(\mu_1-\mu_2)^2}{\alpha_1^{-1}+\alpha_2^{-1}}\right)} \\
&= \mathcal{N}(\mu, \alpha^{-1})\mathcal{N}(\mu_1|\mu_2, \alpha_1^{-1}+\alpha_2^{-1})
\end{aligned}$$

with

$$\begin{aligned}
\mu &= \frac{\alpha_1\mu_1 + \alpha_2\mu_2}{\alpha_1+\alpha_2} \\
\alpha &= \alpha_1 + \alpha_2
\end{aligned}$$

□

Lemma 2.

$$\prod_{i=1}^{n} \mathcal{N}(\mu_i, \alpha_i^{-1}) = c_{1\ldots n}\mathcal{N}(\mu_{1\ldots n}, \alpha_{1\ldots n}^{-1}) \tag{A.2}$$

with

$$\begin{aligned}
\mu_{1\ldots n} &= \frac{\sum_{i=1}^{n}\alpha_i\mu_i}{\sum_{i=1}^{n}\alpha_i} \\
\alpha_{1\ldots n} &= \sum_{i=1}^{n}\alpha_i \\
c_{1\ldots n} &= \prod_{i=2}^{n}\mathcal{N}\left(\mu_i \middle| \frac{\sum_{j=1}^{n-1}\alpha_j\mu_j}{\sum_{j=1}^{n-1}\alpha_j}, \alpha_i^{-1} + \left(\sum_{j=1}^{i-1}\alpha_j\right)^{-1}\right)
\end{aligned}$$

Proof.

$$\prod_{i=1}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = \mathcal{N}(\mu_1,\alpha_1^{-1})\mathcal{N}(\mu_2,\alpha_2^{-1})\prod_{i=3}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

Given lemma A.1

$$\mathcal{N}(\mu_1,\alpha_1^{-1})\mathcal{N}(\mu_2,\alpha_2^{-1})\prod_{i=3}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = c_{12}\mathcal{N}(\mu_{12},\alpha_{12}^{-1})\prod_{i=3}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

with

$$\begin{aligned}\mu_{12} &= \frac{\alpha_1\mu_1+\alpha_2\mu_2}{\alpha_1+\alpha_2}\\ \alpha_{12} &= \alpha_1+\alpha_2\\ c_{12} &= \mathcal{N}(\mu_1|\mu_2,\alpha_1^{-1}+\alpha_2^{-1})\end{aligned}$$

$$c_{12}\mathcal{N}(\mu_{12},\alpha_{12}^{-1})\prod_{i=3}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = c_{12}\mathcal{N}(\mu_{12},\alpha_{12}^{-1})\mathcal{N}(\mu_3,\alpha_3^{-1})\prod_{i=4}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

Again, given lemma A.1

$$c_{12}\mathcal{N}(\mu_{12},\alpha_{12}^{-1})\mathcal{N}(\mu_3,\alpha_3^{-1})\prod_{i=4}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = c_{123}\mathcal{N}(\mu_{123},\alpha_{123}^{-1})\prod_{i=4}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

with

$$\begin{aligned}\mu_{123} &= \frac{\alpha_{12}\mu_{12}+\alpha_3\mu_3}{\alpha_{12}+\alpha_3}\\ &= \frac{(\alpha_1+\alpha_2)\frac{\alpha_1\mu_1+\alpha_2\mu_2}{\alpha_1+\alpha_2}+\alpha_3\mu_3}{\alpha_1+\alpha_2+\alpha_3}\\ &= \frac{\alpha_1\mu_1+\alpha_2\mu_2+\alpha_3\mu_3}{\alpha_1+\alpha_2+\alpha_3}\\ \alpha_{123} &= \alpha_{12}+\alpha_2\\ &= \alpha_1+\alpha_2+\alpha_3\\ c_{123} &= c_{12}\mathcal{N}(\mu_{12}|\mu_3,\alpha_{12}^{-1}+\alpha_3^{-1})\\ &= \mathcal{N}(\mu_1|\mu_2,\alpha_1^{-1}+\alpha_2^{-1})\mathcal{N}(\mu_{12}|\mu_3,\alpha_{12}^{-1}+\alpha_3^{-1})\end{aligned}$$

$$c_{123}\mathcal{N}(\mu_{123},\alpha_{123}^{-1})\prod_{i=4}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = c_{123}\mathcal{N}(\mu_{123},\alpha_{123}^{-1})\mathcal{N}(\mu_4,\alpha_4^{-1})\prod_{i=5}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

Again, given lemma A.1

$$c_{123}\mathcal{N}(\mu_{123},\alpha_{123}^{-1})\mathcal{N}(\mu_4,\alpha_4^{-1})\prod_{i=5}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1}) = c_{1234}\mathcal{N}(\mu_{1234},\alpha_{1234}^{-1})\prod_{i=5}^{n}\mathcal{N}(\mu_i,\alpha_i^{-1})$$

with

$$\begin{aligned}
\mu_{1234} &= \frac{\alpha_{123}\mu_{123} + \alpha_4\mu_4}{\alpha_{123} + \alpha_4} \\
&= \frac{(\alpha_1 + \alpha_2 + \alpha_3)\frac{\alpha_1\mu_1 + \alpha_2\mu_2 + \alpha_3\mu_3}{\alpha_1 + \alpha_2 + \alpha_3} + \alpha_4\mu_4}{\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4} \\
&= \frac{\alpha_1\mu_1 + \alpha_2\mu_2 + \alpha_3\mu_3 + \alpha_4\mu_4}{\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4} \\
\alpha_{1234} &= \alpha_{123} + \alpha_4 \\
&= \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \\
c_{1234} &= c_{123}\mathcal{N}(\mu_{123}|\mu_4, \alpha_{123}^{-1} + \alpha_4^{-1}) \\
&= \mathcal{N}(\mu_1|\mu_2, \alpha_1^{-1} + \alpha_2^{-1})\mathcal{N}(\mu_{12}|\mu_3, \alpha_{12}^{-1} + \alpha_3^{-1})\mathcal{N}(\mu_{123}|\mu_4, \alpha_{123}^{-1} + \alpha_4^{-1})
\end{aligned}$$

and thus

$$\prod_{i=1}^{n} \mathcal{N}(\mu_i, \alpha_i^{-1}) = c_{1\ldots n}\mathcal{N}(\mu_{1\ldots n}, \alpha_{1\ldots n}^{-1})$$

with

$$\begin{aligned}
\mu_{1\ldots n} &= \frac{\sum_{i=1}^{n} \alpha_i\mu_i}{\sum_{i=1}^{n} \alpha_i} \\
\alpha_{1\ldots n} &= \sum_{i=1}^{n} \alpha_i \\
c_{1\ldots n} &= \prod_{i=2}^{n} \mathcal{N}\left(\mu_i \middle| \frac{\sum_{j=1}^{n-1} \alpha_j\mu_j}{\sum_{j=1}^{n-1} \alpha_j}, \alpha_i^{-1} + \left(\sum_{j=1}^{i-1} \alpha_j\right)^{-1}\right)
\end{aligned}$$

□

Lemma 3.

$$p(\mu_x|X) = c^{-1} \sum_{i=1}^{k} \pi_i w_i \mathcal{N}(\Theta_i, \beta_i^{-1}) \tag{A.3}$$

with

$$
\begin{aligned}
\Theta_i &= \frac{\alpha_i \mu_i + \frac{\gamma}{2} \sum_{j=1}^{n} x_j}{\alpha_i + \frac{n\gamma}{2}} \\
\beta_i &= \alpha_i + \frac{n\gamma}{2} \\
w_i &= \mathcal{N}\left(\mu_i \middle| \frac{\sum_{j=1}^{n} x_j}{n}, \alpha_i^{-1} + 2(n\gamma)^{-1}\right) \prod_{j=2}^{n} \mathcal{N}\left(x_j \middle| \frac{\sum_{m=1}^{j-1} x_m}{j-1}, \left(\frac{j}{j-1}\right) 2\gamma^{-1}\right) \\
c &= \sum_{i=1}^{k} \pi_i w_i
\end{aligned}
$$

Proof. By definition of 5.2

$$
\begin{aligned}
p(\mu_x|X) &= p(X|\mu_x)p(\mu_x)c^{-1} \\
&= c^{-1} \sum_{i=1}^{k} \pi_i \mathcal{N}(\mu_i, \alpha_i^{-1}) \prod_{j=1}^{n} \mathcal{N}(x_j, 2\gamma^{-1})
\end{aligned}
$$

Given lemma 2, we can rewrite the product:

$$
\prod_{j=1}^{n} \mathcal{N}(x_j, 2\gamma^{-1}) = c_{1\ldots n} \mathcal{N}(\mu_{1\ldots n}, \alpha_{1\ldots n}^{-1})
$$

with

$$\begin{aligned}
\mu_{1\dots n} &= \frac{\sum_{j=1}^{n} \frac{\gamma}{2} x_j}{\sum_{j=1}^{n} \frac{\gamma}{2}} \\
&= \frac{\frac{\gamma}{2} \sum_{j=1}^{n} x_j}{\frac{n\gamma}{2}} \\
&= \frac{\sum_{j=1}^{n} x_j}{n} \\
\alpha_{1\dots n} &= \sum_{j=1}^{n} \frac{\gamma}{2} \\
&= \frac{n\gamma}{2} \\
c_{1\dots n} &= \prod_{j=2}^{n} \mathcal{N}\left(x_j \middle| \frac{\sum_{m=1}^{j-1} \frac{\gamma}{2} x_m}{\sum_{m=1}^{j-1} \frac{\gamma}{2}}, \frac{2}{\gamma} + \left(\sum_{m=1}^{j-1} \frac{\gamma}{2} \right)^{-1} \right)
\end{aligned}$$

And thus

$$p(\mu_x|X) = c^{-1} \sum_{i=1}^{k} \pi_i \mathcal{N}(\mu_i, \alpha_i^{-1}) c_{1\dots n} \mathcal{N}(\mu_{1\dots n}, \alpha_{1\dots n}^{-1})$$

Given lemma A.1:

$$\mathcal{N}(\mu_i, \alpha_i^{-1}) \mathcal{N}(\mu_{1\dots n}, \alpha_{1\dots n}^{-1}) = w_i \mathcal{N}(\Theta_i, \beta_i^{-1})$$

with

$$\begin{aligned}
\Theta_i &= \frac{\alpha_i \mu_i + \frac{n\gamma}{2} \frac{\sum_{j=1}^{n} x_j}{n}}{\alpha_i + \frac{n\gamma}{2}} \\
&= \frac{\alpha_i \mu_i + \frac{\gamma}{2} \sum_{j=1}^{n} x_j}{\alpha_i + \frac{n\gamma}{2}} \\
\beta_i &= \alpha_i + \frac{n\gamma}{2} \\
w_i &= \mathcal{N}\left(\mu_i \middle| \frac{\sum_{j=1}^{n} x_j}{n}, \alpha_i^{-1} + 2(n\gamma)^{-1} \right) \prod_{j=2}^{n} \mathcal{N}\left(x_j \middle| \frac{\sum_{m=1}^{j-1} x_m}{j-1}, \left(\frac{j}{j-1} \right) 2\gamma^{-1} \right)
\end{aligned}$$

And thus,

$$p(\mu_x|X) = c^{-1} \sum_{i=1}^{k} \pi_i w_i \mathcal{N}(\Theta_i, \beta_i^{-1})$$

□

List of Figures

List of Tables

Bibliography

[1] M. Huerta, G. Downing, F. Haseltine, B. Seto, and Y. Liu, "Nih working definition of bioinformatics and computational biology," *US National Institute of Health*, 2000.

[2] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, *et al.*, "Ncbi geo: archive for functional genomics data sets—update," *Nucleic acids research*, vol. 41, no. D1, pp. D991–D995, 2013.

[3] N. Kolesnikov, E. Hastings, M. Keays, O. Melnichuk, Y. A. Tang, E. Williams, M. Dylag, N. Kurbatova, M. Brandizi, T. Burdett, *et al.*, "Arrayexpress update—simplifying data submissions," *Nucleic acids research*, p. gku1057, 2014.

[4] R. A. Irizarry, D. Warren, F. Spencer, I. F. Kim, S. Biswal, B. C. Frank, E. Gabrielson, J. G. Garcia, J. Geoghegan, G. Germino, *et al.*, "Multiple-laboratory comparison of microarray platforms," *Nature methods*, vol. 2, no. 5, pp. 345–350, 2005.

[5] M. A. Taub, H. C. Bravo, and R. A. Irizarry, "Overcoming bias and systematic errors in next generation sequencing data," *Genome medicine*, vol. 2, no. 12, p. 1, 2010.

[6] K. D. Hansen, Z. Wu, R. A. Irizarry, and J. T. Leek, "Sequencing technology does not eliminate biological variability," *Nature biotechnology*, vol. 29, no. 7, pp. 572–573, 2011.

[7] W. E. Johnson, C. Li, and A. Rabinovic, "Adjusting batch effects in microarray expression data using empirical bayes methods," *Biostatistics*, vol. 8, no. 1, pp. 118–127, 2007.

[8] R. B. Scharpf, H. Tjelmeland, G. Parmigiani, and A. B. Nobel, "A bayesian model for cross-study differential gene expression," *Journal of the American Statistical Association*, 2012.

[9] E. Garrett-Mayer, G. Parmigiani, X. Zhong, L. Cope, and E. Gabrielson, "Cross-study validation and combined analysis of gene expression microarray data," *Biostatistics*, vol. 9, no. 2, pp. 333–354, 2008.

[10] J. K. Choi, U. Yu, S. Kim, and O. J. Yoo, "Combining multiple microarray studies and modeling interstudy variation," *Bioinformatics*, vol. 19, no. suppl 1, pp. i84–i90, 2003.

[11] D. R. Goldstein, M. Delorenzi, R. Luthi-Carter, and T. Sengstag, "Comparison of meta-analysis to combined analysis of a replicated microarray study," *Meta-Analysis and Combining Information in Genetics and Genomics*, vol. 1, pp. 135–156, 2005.

[12] A. A. Shabalin, H. Tjelmeland, C. Fan, C. M. Perou, and A. B. Nobel, "Merging two gene-expression studies via cross-platform normalization," *Bioinformatics*, vol. 24, no. 9, pp. 1154–1160, 2008.

[13] R. A. Irizarry, B. M. Bolstad, F. Collin, L. M. Cope, B. Hobbs, and T. P. Speed, "Summaries of affymetrix genechip probe level data," *Nucleic acids research*, vol. 31, no. 4, pp. e15–e15, 2003.

[14] J. Kim, K. Patel, H. Jung, W. P. Kuo, and L. Ohno-Machado, "Anyexpress: integrated toolkit for analysis of cross-platform gene expression data using a fast interval matching algorithm," *BMC bioinformatics*, vol. 12, no. 1, p. 1, 2011.

[15] K. Engelen, Q. Fu, P. Meysman, A. Sánchez-Rodríguez, R. De Smet, K. Lemmens, A. C. Fierro, and K. Marchal, "Colombos: access port for cross-platform bacterial expression compendia," *PLoS One*, vol. 6, no. 7, p. e20938, 2011.

[16] P. Meysman, P. Sonego, L. Bianco, Q. Fu, D. Ledezma-Tejeida, S. Gama-Castro, V. Liebens, J. Michiels, K. Laukens, K. Marchal, *et al.*, "Colombos v2. 0: an ever expanding collection of bacterial expression compendia," *Nucleic acids research*, p. gkt1086, 2013.

[17] M. Moretto, P. Sonego, N. Dierckxsens, M. Brilli, L. Bianco, D. Ledezma-Tejeida, S. Gama-Castro, M. Galardini, C. Romualdi, K. Laukens, *et al.*, "Colombos v3. 0: leveraging gene expression compendia for cross-species analyses," *Nucleic acids research*, vol. 44, no. D1, pp. D620–D623, 2016.

[18] J. Luo, M. Schumacher, A. Scherer, D. Sanoudou, D. Megherbi, T. Davison, T. Shi, W. Tong, L. Shi, H. Hong, *et al.*, "A comparison of batch

effect removal methods for enhancement of prediction performance using maqc-ii microarray gene expression data," *The pharmacogenomics journal*, vol. 10, no. 4, pp. 278–291, 2010.

[19] C. Camacho, G. Coulouris, V. Avagyan, N. Ma, J. Papadopoulos, K. Bealer, and T. L. Madden, "Blast+: architecture and applications," *BMC bioinformatics*, vol. 10, no. 1, p. 1, 2009.

[20] R. V. Belavkin, J. A. Aston, A. Channon, E. Aston, B. M. Rash, M. Kadirvel, S. Forbes, C. G. Knight, *et al.*, "Mutation rate plasticity in rifampicin resistance depends on escherichia coli cell–cell interactions," *Nature communications*, vol. 5, 2014.

[21] M. Galardini, M. Brilli, G. Spini, M. Rossi, B. Roncaglia, A. Bani, M. Chiancianesi, M. Moretto, K. Engelen, G. Bacci, *et al.*, "Evolution of intraspecific regulatory networks in a multipartite bacterial genome," *PLoS Comput Biol*, vol. 11, no. 9, p. e1004478, 2015.

[22] C. Angione, M. Conway, and P. Lió, "Multiplex methods provide effective integration of multi-omic data in genome-scale models," *BMC bioinformatics*, vol. 17, no. 4, p. 257, 2016.

[23] K. Lemmens, T. De Bie, T. Dhollander, S. C. De Keersmaecker, I. M. Thijs, G. Schoofs, A. De Weerdt, B. De Moor, J. Vanderleyden, J. Collado-Vides, *et al.*, "Distiller: a data integration framework to reveal condition dependency of complex regulons in escherichia coli," *Genome biology*, vol. 10, no. 3, p. 1, 2009.

[24] T. Michoel, R. De Smet, A. Joshi, Y. Van de Peer, and K. Marchal, "Comparative analysis of module-based versus direct methods for reverse-engineering transcriptional regulatory networks," *BMC systems biology*, vol. 3, no. 1, p. 1, 2009.

[25] H. Zhao, L. Cloots, T. Van den Bulcke, Y. Wu, R. De Smet, V. Storms, P. Meysman, K. Engelen, and K. Marchal, "Query-based biclustering of gene expression data using probabilistic relational models," *BMC bioinformatics*, vol. 12, no. Suppl 1, p. S37, 2011.

[26] J. P. Faria, R. Overbeek, F. Xia, M. Rocha, I. Rocha, and C. S. Henry, "Genome-scale bacterial transcriptional regulatory networks: reconstruction and integrated analysis with metabolic models," *Briefings in bioinformatics*, vol. 15, no. 4, pp. 592–611, 2014.

[27] L. Cloots and K. Marchal, "Network-based functional modeling of genomics, transcriptomics and metabolism in bacteria," *Current opinion in microbiology*, vol. 14, no. 5, pp. 599–607, 2011.

[28] M. Kolář, J. Meier, V. Mustonen, M. Lässig, and J. Berg, "Graphalignment: Bayesian pairwise alignment of biological networks," *BMC systems biology*, vol. 6, no. 1, p. 1, 2012.

[29] D. De Maeyer, J. Renkens, L. Cloots, L. De Raedt, and K. Marchal, "Phenetic: network-based interpretation of unstructured gene lists in e. coli," *Molecular BioSystems*, vol. 9, no. 7, pp. 1594–1603, 2013.

[30] Y. I. Balderas-Martínez, M. Savageau, H. Salgado, E. Pérez-Rueda, E. Morett, and J. Collado-Vides, "Transcription factors in escherichia coli prefer the holo conformation," *PloS one*, vol. 8, no. 6, p. e65723, 2013.

[31] M. Moretto, P. Sonego, S. Pilati, G. Malacarne, L. Costantini, L. Grzeskowiak, G. Bagagli, M. S. Grando, C. Moser, and K. Engelen, "Vespucci: Exploring patterns of gene expression in grapevine," *Frontiers in Plant Science*, vol. 7, p. 633, 2016.

[32] "Django web framework." `https://www.djangoproject.com/`. Accessed: 2016-05-30.

[33] R. C. Team, "R: A language and environment for statistical computing. r foundation for statistical computing, vienna, austria. 2013," 2014.

[34] T. Tatusova, S. Ciufo, B. Fedorov, K. O'Neill, and I. Tolstoy, "Refseq microbial genomes database: new representation and annotation strategy," *Nucleic acids research*, vol. 43, no. 7, p. 3872, 2015.

[35] R. P. Huntley, T. Sawford, P. Mutowo-Meullenet, A. Shypitsyna, C. Bonilla, M. J. Martin, and C. O'Donovan, "The goa database: gene ontology annotation updates for 2015," *Nucleic acids research*, vol. 43, no. D1, pp. D1057–D1063, 2015.

[36] H. Salgado, M. Peralta-Gil, S. Gama-Castro, A. Santos-Zavaleta, L. Muñiz-Rascado, J. S. García-Sotelo, V. Weiss, H. Solano-Lira, I. Martínez-Flores, A. Medina-Rivera, *et al.*, "Regulondb v8. 0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more," *Nucleic acids research*, vol. 41, no. D1, pp. D203–D213, 2013.

[37] R. Caspi, T. Altman, R. Billington, K. Dreher, H. Foerster, C. A. Fulcher, T. A. Holland, I. M. Keseler, A. Kothari, A. Kubo, *et al.*, "The metacyc database of metabolic pathways and enzymes and the biocyc collection of pathway/genome databases," *Nucleic acids research*, vol. 42, no. D1, pp. D459–D471, 2014.

[38] I. M. Keseler, A. Mackie, M. Peralta-Gil, A. Santos-Zavaleta, S. Gama-Castro, C. Bonavides-Martínez, C. Fulcher, A. M. Huerta, A. Kothari, M. Krummenacker, *et al.*, "Ecocyc: fusing model organism databases with

systems biology," *Nucleic acids research*, vol. 41, no. D1, pp. D605–D612, 2013.

[39] L. Li, C. J. Stoeckert, and D. S. Roos, "Orthomcl: identification of ortholog groups for eukaryotic genomes," *Genome research*, vol. 13, no. 9, pp. 2178–2189, 2003.

[40] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: a software environment for integrated models of biomolecular interaction networks," *Genome research*, vol. 13, no. 11, pp. 2498–2504, 2003.

[41] M. A. Vivier and I. S. Pretorius, "Genetically tailored grapevines for the wine industry," *Trends in biotechnology*, vol. 20, no. 11, pp. 472–478, 2002.

[42] O. Jaillon, J.-M. Aury, B. Noel, A. Policriti, C. Clepet, A. Casagrande, N. Choisne, S. Aubourg, N. Vitulo, C. Jubin, *et al.*, "The grapevine genome sequence suggests ancestral hexaploidization in major angiosperm phyla," *Nature*, vol. 449, no. 7161, pp. 463–467, 2007.

[43] R. Velasco, A. Zharkikh, M. Troggio, D. A. Cartwright, A. Cestaro, D. Pruss, M. Pindo, L. M. FitzGerald, S. Vezzulli, J. Reid, *et al.*, "A high quality draft consensus sequence of the genome of a heterozygous grapevine variety," *PloS one*, vol. 2, no. 12, p. e1326, 2007.

[44] S. Y. Rhee, J. Dickerson, and D. Xu, "Bioinformatics and its applications in plant biology," *Annu. Rev. Plant Biol.*, vol. 57, pp. 335–360, 2006.

[45] D. C. Wong, C. Sweetman, D. P. Drew, and C. M. Ford, "Vtcdb: a gene co-expression database for the crop species vitis vinifera (grapevine)," *BMC genomics*, vol. 14, no. 1, p. 882, 2013.

[46] A. Pulvirenti, R. Giugno, R. Distefano, G. Pigola, M. Mongiovi, G. Giudice, V. Vendramin, A. Lombardo, F. Cattonaro, and A. Ferro, "A knowledge base for vitis vinifera functional analysis," *BMC systems biology*, vol. 9, no. Suppl 3, p. S5, 2015.

[47] J. Yue, X. Ma, R. Ban, Q. Huang, W. Wang, J. Liu, and Y. Liu, "Fr database 1.0: a resource focused on fruit development and ripening," *Database*, vol. 2015, p. bav002, 2015.

[48] D. C. Wong, C. Sweetman, and C. M. Ford, "Annotation of gene function in citrus using gene expression information and co-expression networks," *BMC plant biology*, vol. 14, no. 1, p. 1, 2014.

[49] Q. Fu, A. C. Fierro, P. Meysman, A. Sanchez-Rodriguez, K. Vandepoele, K. Marchal, and K. Engelen, "Magic: access portal to a cross-platform

gene expression compendium for maize," *Bioinformatics*, vol. 30, no. 9, pp. 1316–1318, 2014.

[50] G. O. Consortium *et al.*, "Gene ontology consortium: going forward," *Nucleic acids research*, vol. 43, no. D1, pp. D1049–D1056, 2015.

[51] M. Kanehisa, Y. Sato, M. Kawashima, M. Furumichi, and M. Tanabe, "Kegg as a reference resource for gene and protein annotation," *Nucleic acids research*, vol. 44, no. D1, pp. D457–D462, 2016.

[52] R. D. Finn, P. Coggill, R. Y. Eberhardt, S. R. Eddy, J. Mistry, A. L. Mitchell, S. C. Potter, M. Punta, M. Qureshi, A. Sangrador-Vegas, *et al.*, "The pfam protein families database: towards a more sustainable future," *Nucleic acids research*, p. gkv1344, 2015.

[53] C. J. Sigrist, E. De Castro, L. Cerutti, B. A. Cuche, N. Hulo, A. Bridge, L. Bougueleret, and I. Xenarios, "New and continuing developments at prosite," *Nucleic acids research*, p. gks1067, 2012.

[54] I. Letunic, T. Doerks, and P. Bork, "Smart: recent updates, new developments and status in 2015," *Nucleic acids research*, vol. 43, no. D1, pp. D257–D260, 2015.

[55] J. Grimplet, G. R. Cramer, J. A. Dickerson, K. Mathiason, J. Van Hemert, and A. Y. Fennell, "Vitisnet:"omics" integration through grapevine molecular networks," *PloS one*, vol. 4, no. 12, p. e8365, 2009.

[56] L. Cooper, R. L. Walls, J. Elser, M. A. Gandolfo, D. W. Stevenson, B. Smith, J. Preece, B. Athreya, C. J. Mungall, S. Rensing, *et al.*, "The plant ontology as a tool for comparative plant anatomy and genomic analyses," *Plant and Cell Physiology*, vol. 54, no. 2, pp. e1–e1, 2013.

[57] B. Coombe, "Grapevine growth stages. the modified el system," *Aust. J. Grape Wine Res*, vol. 1, pp. 100–110, 1995.

[58] J. T. Matus, F. Aquea, and P. Arce-Johnson, "Analysis of the grape myb r2r3 subfamily reveals expanded wine quality-related clades and conserved gene structure organization across vitis and arabidopsis genomes," *BMC Plant Biology*, vol. 8, no. 1, p. 83, 2008.

[59] J. Yin, S. McLoughlin, I. B. Jeffery, A. Glaviano, B. Kennedy, and D. G. Higgins, "Integrating multiple genome annotation databases improves the interpretation of microarray gene expression data," *BMC genomics*, vol. 11, no. 1, p. 1, 2010.

[60] M. Fasoli, S. Dal Santo, S. Zenoni, G. B. Tornielli, L. Farina, A. Zamboni, A. Porceddu, L. Venturini, M. Bicego, V. Murino, *et al.*, "The grapevine expression atlas reveals a deep transcriptome shift driving the entire plant

into a maturation program," *The Plant Cell*, vol. 24, no. 9, pp. 3489–3505, 2012.

[61] S. J. Cookson and N. Ollat, "Grafting with rootstocks induces extensive transcriptional re-programming in the shoot apical meristem of grapevine," *BMC plant biology*, vol. 13, no. 1, p. 1, 2013.

[62] J. Grimplet, J. Van Hemert, P. Carbonell-Bejerano, J. Díaz-Riquelme, J. Dickerson, A. Fennell, M. Pezzotti, and J. M. Martínez-Zapater, "Comparative analysis of grapevine whole-genome gene predictions, functional annotation, categorization and integration of the predicted gene sequences," *BMC Research notes*, vol. 5, no. 1, p. 213, 2012.

[63] L. Costantini, G. Malacarne, S. Lorenzi, M. Troggio, F. Mattivi, C. Moser, and M. S. Grando, "New candidate genes for the fine regulation of the colour of grapes," *Journal of experimental botany*, p. erv159, 2015.

[64] G. Malacarne, L. Costantini, E. Coller, J. Battilana, R. Velasco, U. Vrhovsek, M. S. Grando, and C. Moser, "Regulation of flavonol content and composition in (syrah× pinot noir) mature grapes: integration of transcriptional profiling and metabolic quantitative trait locus analyses," *Journal of experimental botany*, vol. 66, no. 15, pp. 4441–4453, 2015.

[65] P. R. Young, J. G. Lashbrooke, E. Alexandersson, D. Jacobson, C. Moser, R. Velasco, and M. A. Vivier, "The genes and enzymes of the carotenoid metabolic pathway in vitis vinifera l.," *BMC genomics*, vol. 13, no. 1, p. 1, 2012.

[66] J. Grimplet, A.-F. Adam-Blondon, P.-F. Bert, O. Bitz, D. Cantu, C. Davies, S. Delrot, M. Pezzotti, S. Rombauts, and G. R. Cramer, "The grapevine gene nomenclature system," *BMC genomics*, vol. 15, no. 1, p. 1, 2014.

[67] J. G. Lashbrooke, P. R. Young, S. J. Dockrall, K. Vasanth, and M. A. Vivier, "Functional characterisation of three members of the vitis vinifera l. carotenoid cleavage dioxygenase gene family," *BMC plant biology*, vol. 13, no. 1, p. 156, 2013.

[68] X. Yin and P. C. Struik, "Modelling the crop: from system dynamics to systems biology," *Journal of Experimental Botany*, vol. 61, no. 8, pp. 2171–2183, 2010.

[69] B. P. Sheth and V. S. Thaker, "Plant systems biology: insights, advances and challenges," *Planta*, vol. 240, no. 1, pp. 33–54, 2014.

[70] N. Vitulo, C. Forcato, E. C. Carpinelli, A. Telatin, D. Campagna, M. D'Angelo, R. Zimbello, M. Corso, A. Vannozzi, C. Bonghi, *et al.*, "A deep survey of alternative splicing in grape reveals changes in the splicing

machinery related to tissue, stress condition and genotype," *BMC plant biology*, vol. 14, no. 1, p. 1, 2014.

[71] P. M. Lee, *Bayesian statistics: an introduction*. John Wiley & Sons, 2012.

[72] S. G. Johnson, "The nlopt nonlinear-optimization package." `http://ab-initio.mit.edu/nlopt`. Accessed: 2016-05-30.

[73] S. Kauffman, "Homeostasis and differentiation in random genetic control networks," *Nature*, vol. 224, pp. 177–178, 1969.

[74] "Boolean network." `https://en.wikipedia.org/wiki/Boolean_network`. Accessed: 2016-05-30.

[75] S. Gama-Castro, H. Salgado, M. Peralta-Gil, A. Santos-Zavaleta, L. Muniz-Rascado, H. Solano-Lira, V. Jimenez-Jacinto, V. Weiss, J. S. Garcia-Sotelo, A. Lopez-Fuentes, *et al.*, "Regulondb version 7.0: transcriptional regulation of escherichia coli k-12 integrated within genetic sensory response units (gensor units)," *Nucleic acids research*, vol. 39, no. suppl 1, pp. D98–D105, 2011.

[76] C. Müssel, M. Hopfensitz, and H. A. Kestler, "Boolnet—an r package for generation, reconstruction and analysis of boolean networks," *Bioinformatics*, vol. 26, no. 10, pp. 1378–1380, 2010.

[77] R. Van Noorden, "Global scientific output doubles every nine years," *Nature*, 2014.

[78] V. Marx, "Biology: The big challenges of big data," *Nature*, vol. 498, no. 7453, pp. 255–260, 2013.

[79] G. Duck, G. Nenadic, M. Filannino, A. Brass, D. L. Robertson, and R. Stevens, "A survey of bioinformatics database and software usage through mining the literature," *PloS one*, vol. 11, no. 6, p. e0157989, 2016.