Methods for Explaining Biological Systems and High-Throughput Data

Evi Berchtold

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität
München

vorgelegt von
Evi Berchtold
aus München

München, den 07.08.2018

Methods for Explaining Biological Systems and High-Throughput Data

Evi Berchtold

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität
München

vorgelegt von
Evi Berchtold
aus München

München, den 07.08.2018

Erstgutachter: Prof. Dr. Ralf Zimmer
Zweitgutachter: Prof. Dr. Jan Baumbach
Tag der mündlichen Prüfung: 19.10.2018

Eidesstattliche Versicherung

(Siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. .5.)

Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Berchtold, Evi
Name, Vorname

München, 07.08.18
Ort, Datum

Evi Berchtold
Unterschrift Doktorand/in

Formular 3.2

Content

1 Introduction **1**

2 Evaluating Transcription Factor Activity Changes **7**
- 2.1 Introduction . . . 8
- 2.2 Material and Methods . . . 10
  - 2.2.1 Data and networks . . . 10
  - 2.2.2 Unexplained target genes . . . 11
  - 2.2.3 Inconsistency score . . . 13
  - 2.2.4 Optimizing the inconsistency score . . . 14
- 2.3 Results . . . 16
  - 2.3.1 Active TF predictions are very different across methods . . . 16
  - 2.3.2 Performance of methods depends on the particular experiment . . . 17
  - 2.3.3 Assessment of networks . . . 19
  - 2.3.4 Variability of solutions . . . 20
  - 2.3.5 Application to human data . . . 22
- 2.4 Discussion . . . 22
- 2.5 Conclusion . . . 23

3 Comparison of Six Breast Cancer Classifiers using qPCR **25**
- 3.1 Introduction . . . 26
  - 3.1.1 Cancer . . . 26
  - 3.1.2 Breast cancer subtypes . . . 27
  - 3.1.3 Subtype classifiers and risk scores . . . 28
  - 3.1.4 Fluidigm Dynamic Array IFC . . . 29
  - 3.1.5 Prognosis in everyday routine (PiA) study . . . 30
- 3.2 Methods . . . 30
  - 3.2.1 PiA cohort . . . 30
  - 3.2.2 Normalisation . . . 31
  - 3.2.3 Classification . . . 32
  - 3.2.4 Performance and Concordance of Predictions . . . 33
  - 3.2.5 Robustness of Classifications . . . 33
  - 3.2.6 Interactive Report . . . 34
- 3.3 Results . . . 34

3.3.1 Comparability of Fluidigm Chips . . . 34
3.3.2 Survival Analysis . . . 35
3.3.3 Concordance of Classifications . . . 37
3.3.4 Robustness to Noise . . . 39
3.3.5 Interactive Report . . . 40
3.4 Discussion . . . 41

4 RelExplain 43
4.1 Introduction . . . 44
4.2 Methods . . . 46
4.2.1 Network . . . 46
4.2.2 Assigning distances to edges . . . 46
4.2.3 RelExplain Steiner tree approximation . . . 48
4.2.4 Finding alternative paths . . . 49
4.2.5 Evaluation using manually curated subnetworks . . . 50
4.2.6 Application to TCGA data . . . 51
4.3 Results . . . 52
4.3.1 Diauxic shift data . . . 52
4.3.2 Breast cancer data . . . 55
4.4 Discussion . . . 57
4.5 Conclusion . . . 58

5 Modeling of the Changes during Yeast Heat Shock Response 61
5.1 Introduction . . . 62
5.2 Data . . . 63
5.3 Methods . . . 66
5.3.1 Quantitative Modeling . . . 66
5.4 Results . . . 68
5.4.1 Qualitative Modeling . . . 68
5.4.2 Quantitative Modeling . . . 70
5.5 Discussion . . . 77

6 YESdb: Interactive Integrated Analysis of Stress Datasets 81
6.1 Introduction . . . 82
6.2 Data . . . 83
6.2.1 Data search strategy . . . 83
6.2.2 Data processing . . . 84
6.2.3 Data annotation . . . 85
6.3 Workflows . . . 86
6.4 Interactive report . . . 87
6.5 Example analysis . . . 88
6.6 Discussion . . . 91

7 Discussion **93**

Acknowledgements **106**

List of Figures

1.1 Data-Information-Knowledge-Wisdom pyramid 2
1.2 Relationship between the Data-Information-Knowledge-Wisdom pyramid and the *context-level* hierarchy of models 3

2.1 Regulatory effects for all combinations of edge signs and TF activity changes 11
2.2 Discrepancy between active TF predictions of state-of-the-art methods . . 16
2.3 Number of unexplained target genes for the respiratory shift from 20%-oxygen time series and the YEASTRACT and ISMARA networks 17
2.4 Comparison of the number of unexplained target genes for the two alternative definition of unexplained unchanged genes. 19
2.5 Z-score distributions for the comparison of the *i-scores* of the Act-A* solution for the real and randomized Gasch data 20
2.6 Variability of the 10% best Act-A* solutions 21

3.1 Comparison of C(t) values for the RPLP0 gene for 726 samples measured on 10 Fluidigm chips and the CFX platform 35
3.2 Overview of classification results and clinical variables for all patients . . . 37
3.3 Misclassifications per classifier when noise is added 39
3.4 Concordance plot and comparison view in the iReport 40

4.1 Schematic visualization of the RelExplain algorithm 48
4.2 Results of the different methods for the Diauxic,RELEX,YEASTRACT network 55
4.3 Results for the breast cancer data and the ERK1 and ERK2 cascade for RelExplain and SteinerNet 56

5.1 Schematic overview of the changes for the mild and severe heat shock . . . 64
5.2 Half-life and absolute mRNA and protein abundance data used in the quantitative modeling 65
5.3 Comparison of the changes in the different measurements together with the underlying model 69
5.4 Influence of ribosome stalling on the simulation using equilibrium protein synthesis 70

5.5 Influence of protein aggregation on the simulation using equilibrium protein synthesis 71
5.6 Comparison of simulated and measured protein fold changes for the Lee et al. data 72
5.7 Comparison of the simulated and measured protein fold changes at 37°C using different fitting methods 74
5.8 Comparison of the simulated and measured protein fold changes at 42°C using different fitting methods 75
5.9 Comparison of the simulated and measured protein in the independent evaluation test sets 76

6.1 Overview of the data sources and web interface features of YESdb. 84
6.2 Example workflow comparing the consistent genes between heat and salt stress. 88
6.3 An example report with interactive visualizations. 89
6.4 Example report comparing two runs of the example analysis. 90

List of Tables

3.1 Intrinsic breast cancer subtypes, with IHC marker status, grade, outcome and prevalence. 28
3.2 Overview of risk scores. 29
3.3 Clinical characteristics of the PiA cohort, grouped by histopathological subtype. 32
3.4 Logrank p-values, hazard ratios and concordance index for the different risk scores 36
3.5 Cramer's V for risk scores and classifiers 38

4.1 Results of the diauxic shift evaluation for the gluconeogenesis subnetwork . 52

5.1 Overview of simulation results for both 37°C and 42°C 79

6.1 Overview of the annotations of the datasets contained in YESdb. 85
6.2 Overview of workflow transitions. 87

Summary

Explaining biological systems via mathematical models is the goal of most biological high-throughput experiments and the subsequent bioinformatic analyses. The information needed to parameterize a predictive model of a biological system is only rarely available. Instead, different kinds of high-throughput data can be integrated with appropriate bioinformatic methods to construct semi-quantitative or qualitative models, which can explain the data or some aspect of it. Visualization and an interactive analysis of the results necessarily are integral parts of these methods, as they greatly enhance the ability of humans to understand complex relationships.

This thesis describes two different aspects of such bioinformatic methods: the systematic model evaluation together with an interactive visualization of the performance results, and model construction by integrating different kinds of high-throughput data to provide a comprehensive explanation of the system.

For model evaluation, we propose the *i-score* as an independent criterion for evaluation of activity states of transcription factors. It corresponds to the number of target genes whose changes cannot be consistent with the active transcription factors. We found that for most experiments the number of unexplained target genes was huge even if optimized directly, which indicates that the available networks are incomplete and/or contain false edges. An interactive website allows to analyze the changes in the *i-score* if the set of active transcription factors is modified.

Additionally, we provide an interactive visualization of the evaluation of breast cancer subtype classifiers on an independent cohort. Here, it is also possible to analyze subsets of patients or even individual patients which might help to understand the underlying mechanistics of the disease.

RelExplain is a method to construct models by integrating expression data, biological networks and process information to provide a small, understandable subnetwork that best explains a given biological process. In contrast to other significant area search methods the process information is directly taken into account while calculating the subnetwork. This allows to analyze hypotheses about involved processes (e.g. proposed from enrichment methods) in more detail by showing the relationships of genes within the process.

As an example of an integrated analysis of measurements from different experimental techniques we analyzed the transcriptional and translational changes upon yeast heat shock. We modeled the protein abundances given the translational changes to understand the inconsistency between translational and proteomics data.

To facilitate such integrative studies in general, we compiled a database containing experimental data for many different kinds of stress in yeast, that can easily be analyzed using Petri-net based workflows.

Zusammenfassung

Biologische Systeme durch mathematische Modelle zu erklären ist das Ziel der meisten biologischen Hochdurchsatzexperimente und der darauf folgenden bioinformatischen Analyse. Die Informationen, die nötig sind, um ein prädiktives Modell zu parametrisieren, sind allerdings nur selten verfügbar. Stattdessen können verschiedene Arten von Hochdurchsatzdaten mit den entsprechenden bioinformatischen Methoden integriert werden, um ein semi-quantitatives oder qualitatives Modell zu erstellen, das die Daten, oder zumindest einen Aspekt der Daten, erklären kann. Visualisierung und eine interaktive Analyse der Ergebnisse sind notwendigerweise ein integraler Teil dieser Methoden, da sie die Fähigkeit von Menschen, komplexe Zusammenhänge zu verstehen, stark erhöhen.

Diese Arbeit beschreibt zwei verschiedene Aspekte von solchen bioinformatischen Methoden: die systematische Evaluation von Modellen sowie die interaktive Visualisierung der entsprechenden Ergebnisse, und das Erstellen von Modellen durch die Integration von verschiedenen Arten von Hochdurchsatzdaten, um eine umfassende Erklärung des Systems zu liefern.

Zur Modell-Evaluation stellen wir den *i-score* als unabhängiges Evaluationskriterium der Aktivitätszustände von Transkriptionsfaktoren vor. Er entspricht der Anzahl der regulierten Gene, deren Änderung nicht konsistent zu den aktiven Transkriptionsfaktoren sein kann. Wir konnten zeigen, dass diese Anzahl der unerklärten Gene für die meisten Experimente sogar dann hoch war, wenn der *i-score* direkt optimiert wurde. Das weist darauf hin, dass die verfügbaren Netzwerke unvollständig sind und/oder falsche Kanten enthalten. Eine interaktive Website ermöglicht es, die Änderungen des *i-score*s zu analysieren, wenn die Menge der aktiven Transkriptionsfaktoren verändert wird.

Zusätzlich stellen wir eine interaktive Visualisierung der Evaluationsergebnisse von Brustkrebsuntertyp-Klassifikatoren zur Verfügung. Damit ist es möglich, Submengen von Patienten oder sogar einzelne Patienten zu analysieren, was zum Verständnis der zugrunde liegenden Mechanismen der Krankheit beitragen kann.

RelExplain ist eine Methode, um Modelle durch das Integrieren von Expressionsdaten, biologischen Netzwerken und Prozessinformationen zu erstellen. Im Gegensatz zu anderen significant area search Methoden werden die Prozessinformationen direkt mit einbezogen, während das Subnetzwerk berechnet wird. Das erlaubt es, Hypothesen über beteiligte Prozesse (die z.B. von Enrichment-Methoden vorhergesagt wurden) im Detail zu analysieren, indem die Beziehungen zwischen den Genen im Prozess dargestellt werden.

Als ein Beispiel für die integrative Analyse von Messungen mit unterschiedlichen expe-

rimentellen Techniken, haben wir die transkriptionellen und translationellen Änderungen in Hefe während Hitzeschock analysiert. Um die Inkonsistenz zwischen translationellen und Proteome Daten zu verstehen, haben wir die Proteinmengen abhängig von den translationellen Änderungen modelliert.

Um so eine integrative Analyse generell zu vereinfachen, haben wir eine Datenbank erstellt, die experimentelle Daten für verschiedene Arten von Stress in Hefe enthält. Diese Daten können mit einem Petrinetz-basierten Workflowsystem einfach analysiert werden.

Chapter 1

Introduction

> *“Data isn’t information, any more than fifty tons of cement is a skyscraper.”*[91]
>
> **Clifford Stoll**
> **American astronomer and author**

In information science *data*, *information*, *knowledge* and *wisdom* are different, but related concepts that cover different ranges in the ”continuum of understanding”[56]. One of the most common definitions of these concepts is by Ackoff [1]: He defines *data* as symbols that represent properties of entities. *Data* can be gathered and stored (e.g. in databases) but it does not provide meaning in itself without further context [86]. The difference between *data* and *information* is nicely captured in the quote above: *data* is the raw material that has to be processed to be turned into something more useful. The processing of *data* can lead to *information*, that can provide answers to questions of ’who’, ’what’, ’where’, ’when’ and ’how many’. *Information* still represents the properties of entities, but as it contains context it becomes more useful. *Knowledge* provides answers to ’how’ questions which can be given in instructions. It can depend on the experiences of a person and, thus, be personal. In contrast to most other sources [82, 20, 57], Ackoff also defines understanding as answers to ’why’ questions that lie between *knowledge* and *wisdom*. Lastly, *wisdom* is the most personal and vague concept. It cannot be shared like the other concepts, deals with values and while the other concepts are used to understand the present and past, *wisdom* can be used to design the future.

The hierarchical nature of these related concepts is often visualized as a pyramid, called the Data-Information-Knowledge-Wisdom (DIKW) pyramid (see Figure 1.1). The different concepts depend on each other (*information* depends on *data*) and can be transformed into each other (you can acquire *knowledge* through *information*), but the higher these concepts are in the pyramid the more personal they get and can no longer be handled by algorithms.

Most bioinformatic methods provide some kind of model to add semantics to the measured *data* and derived *information*. The most general definition of a model says it is a description of a system, where a system is a set of interrelated objects [45]. A model thus corresponds to *information* or *knowledge* about the system. Similar to the hierarchical structure of the DIKW pyramid, a model can have different levels of detail, i.e. differ-

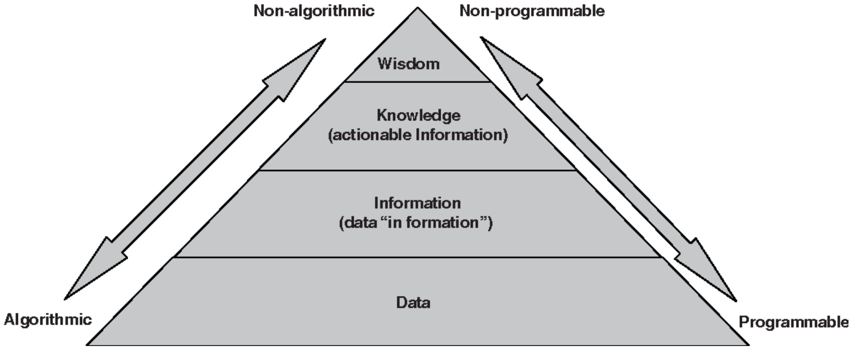


Figure 1.1: Data-Information-Knowledge-Wisdom pyramid. These concepts are hierarchical, as they depend on one another. Algorithms only deal with the first two levels of the pyramid: *data* and *information*. Taken from Awad and Ghaziri [3].

ent *context-levels*. Figure 1.2 shows how the DIKW pyramid and the *context-level* model hierarchy relate to each other. The simplest kind of model describes defined *objects*. Preprocessing methods for raw data typically generate this kind of model as they calculate values/features for each measured gene/protein. Models on the next *context-level* do not only describe *objects*, but also the *relations* between them. Depending on the kind of relations, this can already answer 'how' question and is classified as *knowledge* according to Ackoff. The highest level models contain *functional relations* describing how the *relations* influence the *objects qualitatively* or *quantitatively*. *Wisdom* as defined by Ackoff is a too personal concept to be relevant for the (technical) models discussed in this thesis.

Bioinformatic methods deal with raw data and *information* on different *context-levels* and aim at transforming *data* or models into models on a higher *context-level*. Their use increases both the context of the *information* and enable the understanding of the underlying system. The arrows on the right side of Figure 1.2 next to the bioinformatics methods discussed in this thesis indicate their respective *context-levels*.

Overall, there are four different types of models [45]:

- **conceptual** or **verbal**: a description in natural language describing the system that should be modeled, e.g. the description of the system in a scientific paper
- **diagrammatic**: a graphical representation describing the system by showing the involved entities and their relations, e.g. a Petri net representation of a biological pathway as in KEGG
- **physical**: a real world, physical object often having a different size compared to the original object, e.g. a small scale model of the solar system or a model plane for testing in a wind tunnel
- **formal**: a mathematical model that typically uses equations to describe the behavior of the system, e.g. differential equations describing the changes of metabolites in a pathway

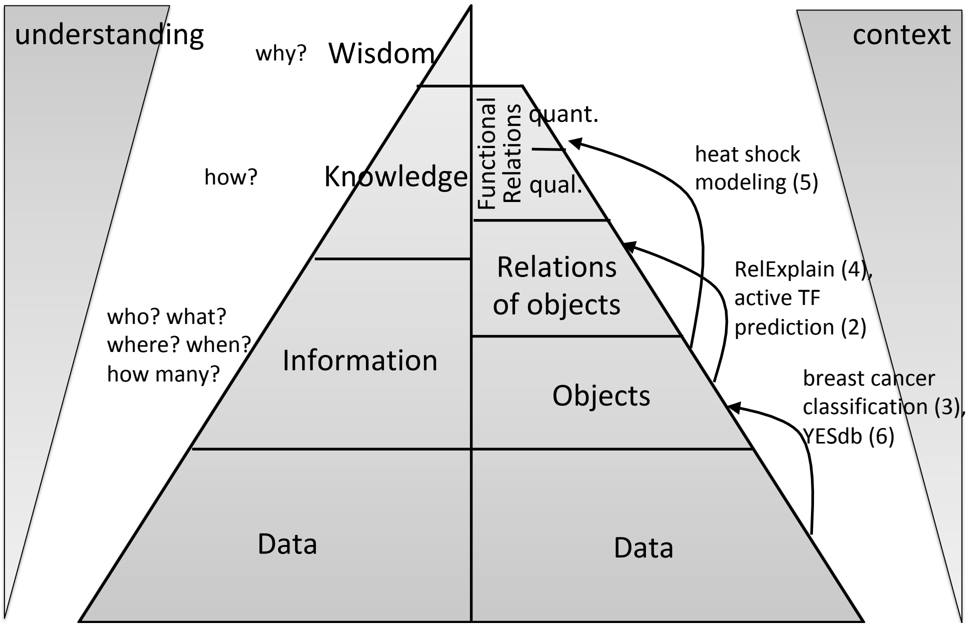


Figure 1.2: Relationship between the Data-Information-Knowledge-Wisdom pyramid and the *context-level* hierarchy of models. The arrows on the right side show how the methods described in this thesis transform models into models with a higher *context-level*. The chapter in which these methods are described are given in parentheses.

Many bioinformatic methods correspond to conceptual, diagrammatic or formal models, but we do not build physical models. Not all types of models can have all *context-levels*. E.g. it does not make sense to have a diagrammatic model describing *objects* without *relations*. But e.g. models with *quantitative functional relations* can have all types, except physical: it can be a verbal/conceptual description, a diagrammatic model with annotated effects on the edges or a formal model.

There are basically three functions of models: (i) to understand the system as e.g. a researcher can gain *knowledge* about the system when studying a diagrammatic schema of the involved processes, (ii) to predict properties of the system e.g. mathematical models can be used to predict the abundances of the involved entities that result from a given initial setting and (iii) to control the modeled system, i.e. to find the parametrization that results in a given output.

Of course, the level of detail of the understanding/prediction/control that can be gained from the model depends on the type and *context-level* of the model. E.g. a conceptual model that describes *relations of objects* will not be able to make quantitative predictions in contrast to a formal mathematical model (i.e. a system of differential equations) that describes *functional relations quantitatively.*

Consider a gene expression measurement. It generates large amounts of *data*: for each gene at least one numeric value, with replicates and multiple conditions even several values per gene. These values describe how much mRNA of the gene is available in the cell, but without further context they are not very useful - they correspond to (raw) *data*, not a model or *information*. Bioinformatic methods to determine differential expression of genes can convert this *data* to a model describing *objects* by determining 'how many' and 'which' genes are changing between conditions. This *information* is still only a crude description of the properties of the system.

To understand 'how' the biological system behaves, models with a higher *context-level* are required. Bioinformatic methods such as the prediction of active transcription factors can be used to come up with a hypothesis of which transcription factors influence the changing genes. This is a model describing the *relations of objects* which makes testable predictions. To validate or falsify this hypothesis further experiments are needed.

Systems biology aims at formal models describing the *quantitative functional relations* of overall complete systems. However, the information needed to construct such a systems biology model is often not available for the biological system of interest. E.g. if one is interested in the transcriptional regulation of a system, one needs not only to find the involved transcription factors, but also how they interact and which combinations of active transcription factors regulate which genes. To acquire this information it is not sufficient to identify the involved transcription factors by ChIP-seq or a related technique and to analyze the knock-out mutant of each of the involved transcription factors (which already requires a large number of experiments), but in order to understand the combinatorics of the transcription factors multi knock-out mutants have to be measured. Such specific experiments usually cannot be taken from available large scale studies but need to be done individually for each biological system.

Even though these specific experiments are often not available, for many model organisms quite a lot of information on transcriptional regulation is available. Large scale transcription factor binding and knock-out studies can be used to generate binary regulatory networks, in which each edge corresponds to an activating or inhibiting regulation of a transcription factor and a target gene. This information is not sufficient to construct a model that can predict the gene expression changes (e.g. a Petri net with elaborate firing rules), but such a binary network can be used for many bioinformatic methods such as predicting which transcription factors are active. The information gained from these methods can then be used to understand the regulation of the system to some extent. Similarly, for many different aspects of a system large scale systematic data is available that can help to understand at least parts of the system.

A good model integrates as much available information as possible. This allows a holistic understanding of the system and prevents that already available data contradicts the model. There are different types of integrative models: A model can integrate different types of data/information, e.g. data from different experimental techniques and additional information from different sources. Another type of integration is to integrate data of the same type that was measured multiple times, e.g. in different individuals or by different platforms or labs. Both types of integration can make the model more robust to noise as the conclusions typically depend on several measurements.

Another aspect that is vital to understand complex biological mechanisms are good visualizations. A diagrammatic model is much easier to understand for the human mind than a corresponding conceptual model in natural language. While integration of multiple data sources typically increases the model quality, it makes the visualization more complex, as e.g. multiple measurements of the same entity have to be summarized. As a consequence, information that can be crucial to evaluate a model is not available in the visualization. Take for example a regulatory network showing transcription factors and their corresponding target genes for some biological system, in which the changes of both transcription factors and target genes are color-coded. When measurements for the changes of multiple individuals are available and the average change is shown in the visualization, the information about how consistent the measurements are between the different individuals and how consistent the depicted regulations are in the individual measurements is lost. One solution is to provide an interactive visualization that can show the corresponding individual measurements. Such interactive visualizations allows e.g. to zoom in to a part of a model and show this section in more detail. Similarly, external resources can be linked to provide more information about an entity or the source of the depicted information.

Once a model has been created, it is important to evaluate its performance to make sure it is consistent with the data. However, in many cases the truth is unknown so that it is impossible to create a gold standard against which the output can be compared. In these cases the best option is to analyze whether the predicted solution shows the same properties that the 'true' solution is expected to exhibit. In any case it is important to use independent test sets to prevent overfitting of the model.

This thesis deals with different methods to make use of the available high-throughput data to better understand and, thus, explain biological systems. Two different aspects are covered: The first two chapters evaluate the performance of different published methods and the next three chapters present methods to create different types of models facilitating the step from *information* to *knowledge* or *information* with higher-level context (see Figure 1.2 for an overview).

In case of the prediction of active transcription factors there is no 'gold standard' solution against which a given prediction can be compared. To nevertheless compare the performance of various published methods to predict the active transcription factors we propose the inconsistency score (*i-score*) in Chapter 2. The true set of active transcription factors should explain all the observed changes. The *i-score* basically calculates how many target genes are strictly inconsistent with the predicted active transcription factors, so that no (reasonable) combinatorics of the transcription factors can yield the observed changes. It uses an approach that is completely different from the methods that are evaluated and thereby provides an independent comparison of the methods. It also does not require the quite unrealistic assumption that the effects of the different transcription factors sum up to the expression changes of the target genes, while still holistically integrating the predicted activities of all transcription factors at once. Even though it generally underestimates the number of inconsistent target genes, we found that most datasets yield many inconsistent target genes even when the *i-score* was optimized directly. This indicates that the gene regulatory networks that are used as input are still incomplete.

Chapter 3 compares different classifiers that predict the risk of recurrence for individual breast cancer patients using gene expression measurements of several genes. For this kind of methods there are standard evaluation methods such as Kaplan-Meier plots and log-rank p-values used in survival analysis. We showed that it is possible to measure the gene expression using a mid-throughput qPCR platform, which enables the qPCR measurements of many genes for many patients in a standardized way without much effort. Most of the evaluated classifiers performed well in this evaluation setup that did not only use an independent cohort but also a different experimental platform. This experimental technique makes such meta-studies on completely independent new cohorts more feasible and can thus contribute to more comparability between the classifiers

The next three chapters describe methods that create models that integrate different kinds of multi-omics data to explain and, thus, understand some biological system. The resulting models contain different levels of detail and deal with different aspects of the system.

In Chapter 4 transcriptomics data is integrated with network data and process information to construct diagrammatic models. Biological networks contain information about the relations between genes or proteins, such as regulation or interaction. The amount of information contained in networks is huge, with hundreds of thousands of edges in some networks. Thus, to not return incomprehensible hairball networks it is important to focus on the most important (i.e. most deregulated) subnetworks. To understand such small subnetworks an enrichment analysis of the contained genes is often done to determine in which process this subnetwork is involved in. RelExplain works the other way round and returns the optimal subnetwork for a given biological process of interest, so that this information can be exploited already during the optimization. As the resulting subnetworks take this additional context into account they can be understood more easily.

Chapter 5 analyses the post-transcriptional regulation of a biological system - the yeast heat shock response. Specifically, it is analyzed how the changes that are observed on the expression level are passed on to the protein levels. This is analyzed using both qualitative and quantitative models that predict the expected protein abundance given the changes in translation.

More generally, Chapter 6 describes the YESdb a database of all available high-throughput experiments of stress response in yeast. The datasets in this database are annotated with the type, duration and strength of the applied stress, the experimental platform that was used and when the corresponding paper was published. Using these datasets Petri net like workflows can be created that define and characterize sets of genes that are interesting in multiple datasets. The results of these workflows can be visualized in interactive reports that can also be shared.

In conclusion, the methods presented in this thesis highlight how the integration of data can improve the understanding of biological systems, especially when combined with interactive visualization.

Chapter 2

Evaluating Transcription Factor Activity Changes

Motivation

The evaluation of the performance of a method is crucial to show the validity of the approach and to compare it to different approaches. Independent test sets and/or independent evaluation measures are used to ensure that the performance is not overestimated. In the next two chapters we describe two evaluation approaches, one using an independent test set and one using additionally an independent evaluation measure.

Currently, there is no experimental method to measure the activity states of all transcription factors (TFs). There are several methods to predict from a regulatory network and expression data which TFs are active, but there is no evaluation measure. Here, we present such an evaluation strategy that indicates for how many target genes the observed expression changes can be explained by a given set of active TFs. As none of the tested methods optimize this measure directly it can be used as independent measure to evaluate these methods. To overcome the problem that the exact combination of active TFs needed to activate a gene is typically not known, we assume a gene to be explained if there exists *any* combination for which the predicted active TFs can possibly explain the observed change of the gene. We introduce the *i-score*, which quantifies how many genes could not be explained by the set of active TFs.

We observe that, even for these minimal requirements, published methods yield many unexplained target genes, i.e. large *i-scores*. This holds for all methods and all expression datasets we evaluated. We provide new optimization methods to calculate the best possible (minimal) *i-score* given the network and measured expression data. The evaluation of this optimized *i-score* on a large data compendium yields many unexplained target genes for almost every case. This indicates that currently available regulatory networks are still far from being complete. Both the presented Act-SAT and Act-A* methods produce optimal sets of TF activity changes, which can be used to investigate the difficult interplay of expression and network data.

Publication

The content of this chapter was published in PLOS ONE ([8]). Here, the manuscript is reformatted and parts of the Supplement are integrated.

Author Contributions

Evi Berchtold analyzed the data, implemented and evaluated the method and wrote the paper. Evi Berchtold and Gergely Csaba designed the method. Gergely Csaba implemented a prototype of the A* approach. Ralf Zimmer supervised the project and edited the paper.

Availability

A web server and a command line tool to calculate our *i-score* and to find the active TFs associated with the minimal *i-score* is available from

`https://services.bio.ifi.lmu.de/i-score`

2.1 Introduction

The goal of many high-throughput experiments is to derive models of regulatory mechanisms that explain the observed changes. For gene expression measurements, such as microarray and RNAseq measurements, the differential activation of transcription factors (TFs) can be considered a major cause for the observed differential expression of the measured genes.

While there are established methods to measure the mRNA level there is no experimental high-throughput method that can determine which TFs are active, i.e. actually regulate their target genes in the current context. Two experimental approaches are used to infer such activities: ChIP and perturbation (knock-out/-down) experiments. ChIP experiments can, for one TF at a time, determine the binding sites of a TF, whereas knock-out experiments measure affected genes following an elimination, deactivation, or perturbation of one or several TFs. For the purpose of deriving TF activity changes these experimental methods have a number of disadvantages: (i) Binding of a TF to the promoter of a gene alone is not always enough to regulate the gene, as post-translational modifications may be needed to activate the bound TF. (ii) Even if the binding of the TF has an effect on the target gene, it is not clear whether the expression will be up- or downregulated by the bound TF. (iii) Multiple TFs can regulate a gene and it is often unknown whether and how they have to interact to affect gene expression. It is also unclear what the overall effect is if some TFs are activating and others inhibiting. (iv) ChIP experiments are usually not available for all TFs for the experimental conditions analyzed. (v) They are rarely done differentially between two conditions to compare whether the binding of a TF changes. (vi) It is unclear whether the observed changes are a direct consequence of the knocked-out TF or of some downstream regulatory cascade or due to of a side effect of the perturbation itself.

Both types of experiments are context-specific, both with respect to the specific bindings and in particular to the effects on possible target genes. Instead of doing differential experiments to uncover which TFs bind in the specific setting, experiments in many conditions can be used to compile gene regulatory networks that indicate which targets may be regulated by which TF in at least one condition. If both knock-out and ChIP experiments are available it is additionally possible to assign a sign to the regulation, that is whether the target gene is up- or downregulated by the TF. Again, the regulation and its sign are likely context-specific.

A goal of many experiments is to uncover regulatory mechanisms that explain the observed differences between the analyzed conditions. Active TFs are the first regulative layer of gene expression and, as the results of the differential activation (i.e. the transcriptional gene expression) are measured directly, can be analyzed more easily than other regulatory mechanisms.

To analyze which TFs are differentially active, computational methods are needed that predict from gene regulatory networks and (a set of) transcriptional measurements the actual activity changes of relevant TFs. This task is challenging as the networks are not complete and, on the other hand, contain many condition-dependent regulations. Furthermore, if a TF is differentially active it does not necessarily regulate all its annotated target genes. Some genes only change their expression if several TFs are active and interact. The precise combination of TFs that have to be active to change the target gene's expression is rarely known.

Nevertheless, there are several methods which predict the activity changes of TFs or activities from which the changes can easily be calculated.

Bussemaker [16] introduced a method that models gene expression as an additive combination of TF activities. A matrix A that contains the gene expression of several conditions and a matrix F that gives the effect strength for each TF-target combination are employed. Multivariate linear regression is used to infer the activities of the TFs T via the equation $A = C + FT$, where C is a constant matrix. There are several variants of this approach that differ in the way the TF activities are derived and whether motif occurrence and/or ChIP data is used for the effect strengths of the TFs in F. We focus here on *ISMARA* [5] and *plsgenomics* [14] as they are available for assessment as a webserver or as R-package, respectively. The R-package *plsgenomics* takes ChIP data or binary network information for F and uses partial least square regression to infer the TF activities. *ISMARA* takes motif occurrences for F and is available as a webserver. In addition to solving the undisturbed model, it also solves an *in-silico* knock-out model for each TF by removing all regulations of the TF from F. The difference between the normal and the knock-out model is used to calculate a z-score that indicates how important the respective TF is for the experiments.

T-profiler [13] performs a t-test between the fold changes of the genes that are targets of a TF and all other genes. To account for overlaps in the target sets of the TFs, the method iteratively selects the TF with the best p-value and then subtracts the mean expression of the genes in the target set from all genes in this target set. An advantage of T-profiler is that it only needs a gene regulatory network and one dataset of fold changes as compared to the many conditions needed for other methods.

DREM [28] uses time series expression data and ChIP data to cluster genes to bifurcating paths of expression changes over time. A bifurcating path indicates that the genes had similar expression values up to this point but diverge systematically afterwards. These bifurcations are then explained by active TFs.

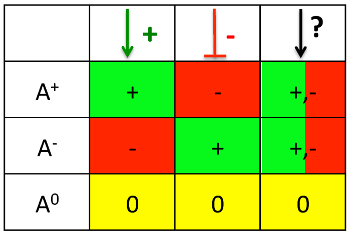
The assessment of the performance of these methods is *generally* very difficult as no gold standard for neither networks nor data nor true activities for TFs are available. To nevertheless systematically evaluate such methods we propose an evaluation score, called the inconsistency score (*i-score*), which indicates, for a given regulatory network, how well the observed changes of the target genes can be explained by a given prediction of differentially active TFs. More specifically it measures the weighted number of gene expression changes, which can **not** be explained by the set of activity changes of TFs in question. This *i-score* is easy to interpret and as it is not optimized directly by any of the methods the *i-score* is well suited to compare and assess their predictions. To compute the *i-score* only a list of TF activity changes, fold changes and the gene regulatory network are needed as input. Thus, results of all mentioned methods can easily be evaluated.

In addition, we provide two methods to obtain the set of differentially active TFs that achieves the best *i-score*. The first method Act-SAT is based on a max-SAT solver and computes the globally best set of activity changes. The second method Act-A* is based on the A* algorithm [50] and computes all optimal solutions which involve only a predefined small number of differentially active TFs. In any case, these optimized *i-scores* constitute the respective theoretical minima given the network and data. These minima can be compared to the *i-scores* of the predictions of various methods to assess how far they are from the optimum. Surprisingly, even if the *i-score* is optimized directly, it is not possible to explain all observed changes. Due to errors in the network or noise in the data many target genes remain unexplained even for the optimal set of activity changes. On the other hand, our Act-SAT and Act-A* methods yield optimal sets of activity changes of TFs explaining most of the observed expression changes. A* delivers such sets with only few differentially active TFs, which are easy to interpret and to use in subsequent analyses and validations. Moreover, the set of unexplained target genes and inconsistent edges might constructively hint to interesting hypotheses implied by the actual data (based on the given regulatory network).

2.2 Material and Methods

2.2.1 Data and networks

For our evaluation we applied the different methods to three datasets and two networks. Our method can be applied to any organism. In this paper, we focus on yeast as with YEASTRACT [93] a large regulatory network of good quality is available which can serve as a kind of common gold standard for all the methods. The YEASTRACT network



	↓+	⊥-	↓?
A^+	+	-	+,-
A^-	-	+	+,-
A^0	0	0	0

Figure 2.1: Regulatory effects for all combinations of edge signs (columns) and TF activity changes (rows). E.g. an activating edge (first column) has an activating regulatory effect (+) on the target gene if the associated TF is more active (A^+). If the sign of the edge is not known (last column) the regulatory effect can be assumed to be activating or inhibiting depending on which is needed to explain the target gene.

comes close to such a standard for yeast, while in most other organisms the situation can be expected to be much worse, i.e. more error-prone, more context-dependent and much more incomplete. In addition to YEASTRACT that contains only experimentally validated regulations, we also include the more putative and much larger motif-derived network used by ISMARA. In this network an edge indicates that the binding motif of a TF matches the promoter of the target. Again, for yeast such a network is more reliable as in other species due to the available data and the assumed complexity of the regulatory mechanisms.

Furthermore, as baseline for comparison, we constructed our own motif-based network using the TF binding motifs provided by Jaspar [83]. We created two different networks by using the MEME suite [4] to search for binding motifs in the region 250 bp and 1000 bp upstream of the TSS for all yeast genes. The network constructed from the 250 bp upstream of the TSS (called Jaspar 250) contains 40.441 edges and is comparable in size to YEASTRACT (41.498 edges) and the other network (called Jaspar 1000) contains 146.431 edges and its size is comparable to ISMARA (155.404 edges).

As experimental data we used a time series that analyzed the transition of respiratory and respirofermentative cultures to fully fermentative metabolism by monitoring the changes of yeast cultures grown initially with 1% and 20.9% oxygen, respectively, after transition to 0% oxygen [81]. We also used other datasets, not discussed in this paper, and the results are very similar (see Supplement).

Furthermore, to assess the influence of the network systematically we compared the performance for real and randomized data in a large compendium containing many experiments. For this we employed the compendium by Gasch [37] with differential data for 173 experiments measuring the reaction of yeast to several environmental stresses.

2.2.2 Unexplained target genes

In order to predict the TF activities, the available methods have to make strong assumptions. The regression model used by ISMARA and plsgenomics assumes that the measured expression levels/fold changes are linear combinations of the TF activities. However, it is known that the effects of TFs do not have to be additive and it is possible that a TF can regulate its target gene only if another TF is also active. Thus, if only one of the two TFs is active the expression of the target gene might not change at all. In this case the expression predicted by the regression model will be far from the observed expression as the model is not suitable for this kind of TF-TF interaction, i.e. this particular activation function. For regression models a natural measure to assess the prediction are residuals (the fitting errors per gene). But because of these too restraining assumptions of the model it is not very meaningful to use the residual of the regression fit as a measure of how good the predicted TF activities explain the observed effects. A high residual could either be due to such non-additive effects or due to falsely predicted TF activities.

Here, we propose a more realistic model [64], which needs to be much more general. It is based on Petri nets to model several regulators which could cooperate according to a general activation function to regulate the target gene. This function can depend on binding strength, activity changes, protein concentrations, etc. but is abstracted here as a (maybe complicated) function of the activity changes of the regulating TFs. Such a model may be realistic but, of course, it is not available and cannot directly be used to assess the performance of other (simpler or even more complicated) models. Therefore, rather than formulating a specific model, we introduce the notion of **unexplainability** in order to measure whether activity changes (predictions) cannot explain the data for *any* reasonable activation function. For an explained gene the actual activation function could still be such that the predicted activity changes do not explain the observed effect. Thus, the unexplainability of activity changes of TFs given a regulatory network and data yields a lower bound of the defects (data not explained) of TF activity change predictions. The unexplained activity changes are either wrong, or more and other regulators are required.

As we want to analyze *differential* experiments, we define three activation changes for the TFs. If a TF is similarly active in both conditions we define it to have unchanged activity (A^0). If it is differentially active, it can either be more active (A^+) or less active (A^-). In the following, an active TF always means a differentially active TF.

The **regulatory effect** is the direction of the expected change of the target gene given the annotated sign of the edge (+/-) and the activity change of the TF. A TF can have an activating (+), inhibiting (-) or no (0) regulatory effect on the target gene. Figure 2.1 shows for all combinations of edge signs and TF activity changes the resulting regulatory effect. If no sign is annotated to the edge, the effect can be either activating or inhibiting, depending on the actual sign of the edge. But as this sign is unknown, we can optimistically assume that the sign is such that the regulatory effect explains the target gene if possible.

A **target gene is unexplained** if for none of its associated TFs the regulatory effect and the target gene's change are in the same direction.

1. If the target gene is differential (changed) it is sufficient that at least *one* of its TFs is predicted to have the corresponding regulatory effect: as the activation function is unknown, the changing TFs could cause the change of the target (e.g. for an activation function that combines the regulatory effects in an OR-like way).

2. An unchanged target gene is explained if there is at least one TF with no regulatory effect (i.e. one unchanged TF (A^0)). The unchanged TF could be the reason that the

target gene is not changed, as the TF is essential for a change (e.g. for an AND-like activation).

Theoretically, it is possible that there are complementary TF activity configurations (i.e. all TFs change their activity) with the same overall effect on the target gene. E.g. if there are two TFs A and B and the target gene is expressed at the same level if either one is active and the other inactive, the target gene can also be unchanged between two conditions if A and B both change but have opposite regulatory effects on the target gene so that they cancel each other out. Strictly, an unchanged target gene could thus only be unexplained if all its TF show the same regulatory effect.

However, we assume that this is rarely the case, as the activity change of both TFs would have to result in the exact same expression change on the target gene so that no expression difference between the two analyzed conditions is observed. Thus, we want to minimize these cases and count the target gene as unexplained if none of its associated TFs are predicted to have unchanged activity (A^0) disregarding complementary TF activity configurations. In addition, we analyzed the effect of these complementary TF activity configurations and found that many unexplained target genes remain unexplained even with this alternative definition.

2.2.3 Inconsistency score

If a given prediction of activity changes of TFs is correct one would expect no or only very few unexplained target genes (UTG). The number of unexplained target genes (#UTG) is thus a suitable measure to assess predictions of active TFs. As fold changes of unexplained genes may differ a lot, a weighted inconsistency score might be an even more appropriate measure for the quality of the activity change predictions.

For target genes with a fold change close to the differential cutoff c one is less certain whether it is really differential or not, while genes with fold changes far from the cutoff are more certain. This can be taken into account by using a score that incorporates the log fold change of the UTG. The inconsistency score (*i-score*) of a given set of active TFs is calculated as the differences of the log fold change fc_t of the target gene t to the cutoff c, summed for all UTGs. As differential target genes can have a much larger difference to the cutoff than unchanged genes, the fold changes are trimmed to a maximal log fold change m_{fc}.

$$\textit{i-score} = \sum_{t \in \text{UTG}} |\min(fc_t, m_{fc}) - c| \tag{2.1}$$

This is of course only one possible way to score the UTGs. One could as well weight differential target genes differently or score only a subset of all genes. The latter may be useful if the user is interested in certain sets of signature genes, pathways and/or GO categories. For our evaluation in the following we use the *i-score* (2.1) as it constitutes a good balance of the overall inconsistency of all network regulations and the individual differential and unchanged target genes.

2.2.4 Optimizing the inconsistency score

We applied the *i-score* to the predictions of different methods for several datasets in order to compare their performance. Surprisingly, all methods yield a rather large number of UTGs for most conditions and, thus, a high *i-score*. In order to assess whether the unexplainability is due to the incompleteness and context-specificity of the network or due to noise in the data and not just poor predictions, we calculate the set of active TFs that yields the minimal *i-score*. This minimal *i-score* is the theoretical optimum of the unexplainability given the network and the data.

In the following two optimizations are introduced: one that optimizes the unexplainability without any further restrictions (Activity SAT or Act-SAT) and one that limits the number of active TFs (Act-A*). The second variant is probably more realistic, as one is typically interested in the most important TFs, and a solution in which a large fraction or all TFs are predicted to be active is often meaningless and useless for follow-up experiments. As we employ the optimal *i-score* for assessment and as a comparison of the *i-score* of the actual prediction, even suboptimal scores are useful as lower bounds. Note that even though we optimize the *i-score*, we also report the corresponding #UTG. Both scores together provide a better interpretation as one can assess how many genes are unexplained and how far they are from the fold change cutoff on average.

Act-SAT

The optimization of the *i-score* can be modeled as a weighted max-SAT problem [58] and then solved by a weighted max-SAT solver, e.g. akmaxsat [63] or an incomplete weighted max-SAT solver e.g. Dist [17]. While a complete SAT solver guarantees to find the optimal solution, but might take very long, an incomplete SAT solver aborts the optimization after a given time and returns the best solution found so far. Given a SAT formula with weights for each clause, these solvers find the solution with minimal weight for the unfulfilled clauses and, thus, the minimal *i-score*. For our optimization the weighted SAT is given in conjunctive normal form (CNF) as follows. There are three variables for each TF_i: one that indicates that the TF is more active (A_i^+), one for less active (A_i^-) and one for a neutral TF (A_i^0). For each target gene g one clause is added to the formula:

unchanged target: $\bigvee_{i \in TF} A_i^0$
upregulated target: $\bigvee_{i \in act} A_i^+ \vee \bigvee_{i \in inhib} A_i^-$
downregulated target: $\bigvee_{i \in act} A_i^- \vee \bigvee_{i \in inhib} A_i^+$

For unchanged target genes the unchanged activity variable of all its associated TFs are combined by OR. For up-/downregulated genes it depends on the sign of the edge which variable is used. For upregulated genes at least one of all TFs with an activating edge to the gene (*act*) has to be more active (A^+) or at least one TF with an inhibiting edge (*inhib*) has to be less active (A^-). Edges for which it is not known whether they are activating or inhibiting are treated as both, so that these TFs are contained in *act* and *inhib*. Note that additional edges (missing in the current network) can only decrease the *i-score* as they imply additional literals in clauses which could satisfy them and thereby reduce the number of unfulfilled clauses.

The weight of each clause is the score that this gene would yield if it is unexplained. The max-SAT solver then finds a solution such that the sum of the weights of the unfulfilled clauses is minimized.

Furthermore, in a valid prediction only one of the three state variables of a TF has to be set to true. Therefore, for each TF the following four clauses are added as *hard* clauses to the SAT formula i.e. they are given a weight that is higher than the sum of all the (soft) target gene clauses. Thus, a solution for which for all TFs exactly one of the state variables is true, always scores better than a solution where not exactly one of the three state variables is true. The first of these clauses guarantees that at least one of the three states is true, and the other clauses guarantee that it is not possible that two states are true at the same time.

$$(A^+ \vee A^- \vee A^0) \wedge\ (\neg A^+ \vee \neg A^-) \wedge\ (\neg A^+ \vee \neg A^0) \wedge\ (\neg A^- \vee \neg A^0)$$

Act-A*

Even though SAT solvers are fast for most problem instances it is possible that it takes impractically long to obtain the optimum (the problem is NP hard!). Moreover, if further constraints should be used in the optimization it is usually not straightforward to encode them in the SAT formula. E.g. as it is unlikely that very many TFs are changing their activity it is reasonable to limit the maximal number of (differentially) active TFs, but it is not straightforward to modify the SAT formula to incorporate this constraint.

Therefore, we use a more flexible optimization method based on the A* informed search algorithm [50]. Act-A* iteratively extends partial solutions until all relevant complete solutions are created. It can be used to find the best solution with at most N active TFs. A partial solution contains less then N active TFs. The search starts with the partial solution with no active TFs, and in each extension step one of the not yet active TFs is set to the more active (A^+) or less active state (A^-).

To enable an informed search by Act-A* we have to estimate partial solutions by an admissible, i.e. optimistic heuristic. In each expansion step the *i-score* of the partial solution is estimated by the admissible heuristic, and if it is worse than the best score of a complete solution already obtained, the partial solution is no longer extended. As the real score of this partial solution is always worse than the heuristic score, optimal solutions are never discarded, but many suboptimal solutions will be skipped.

To calculate the optimistic heuristic score for such a partial solution containing x active TFs, we first calculate the (normal) *i-score* of this solution. For each of the not yet set TFs the improvement of this score is calculated, for the two cases where the TF is set to the more (A^+) or less (A^-) active state. These score improvements are sorted and the first $N - x$ of them are subtracted from the original score. This is an optimistic estimate as the score improvements will decrease with each TF that is set, as target genes are already explained by the set TF and can no longer be explained by other TFs.

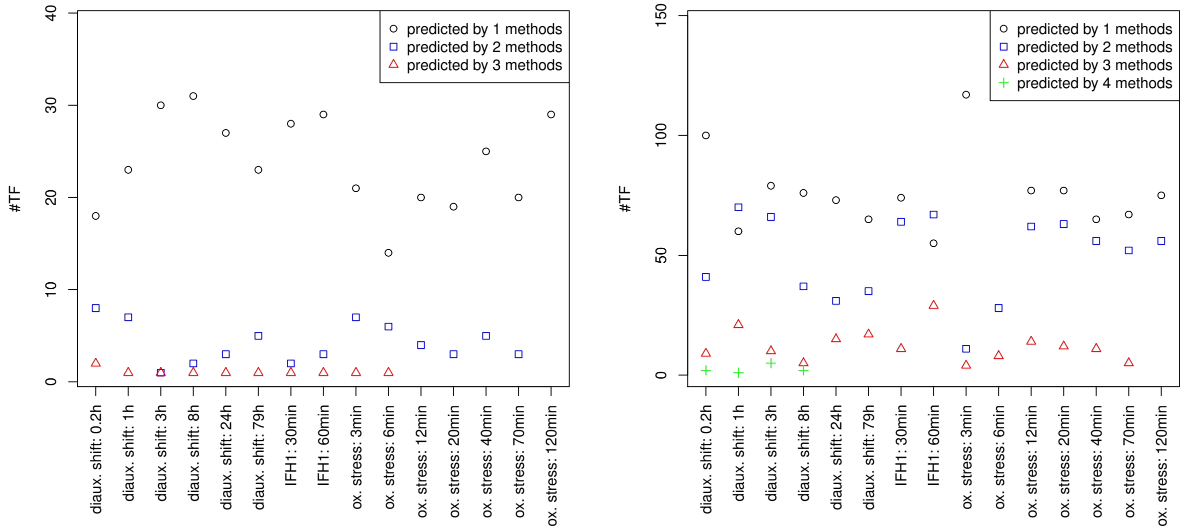


Figure 2.2: Discrepancy between state-of-the-art methods. For each condition in the selected datasets the number of TFs that were in the top 10 TFs (left) and all predicted TFs (right) for four different methods (ISMARA, plsgenomics, DREM, T-profiler) are plotted. Most TFs are only predicted by one of the methods. There is no TF that was predicted to be active by all methods if only the top 10 TFs were used and only very few if all predicted TFs are considered.

2.3 Results

2.3.1 Active TF predictions are very different across methods

To assess how divergent the predictions of the different methods are, we analyzed how many TFs are predicted by various methods. For each condition of the selected datasets the differential activity of TFs are predicted by all methods and for each TF it is assessed by how many methods it is predicted. This can also be restricted to the most important TFs by restricting the prediction to the top (most changing) 10 TFs. The left part of Figure 2.2 shows how many TFs are predicted to be in the top 10 TFs by 1, 2, 3 or 4 methods. Surprisingly, there was *no* TF commonly predicted by all 4 methods, only few that are predicted by 3 methods (red triangles) and most TFs are predicted by only one method (black circles). If the unrestricted prediction is assessed (right part of Figure 2.2) the results are similar and only very few TFs are predicted by all 4 methods.

Thus, the resulting activity changes strongly depend on the used method. Different methods are not even consistent with respect to the most changing TFs. As a consequence, it is especially important to be able to assess which method performs well for a given combination of data and network.

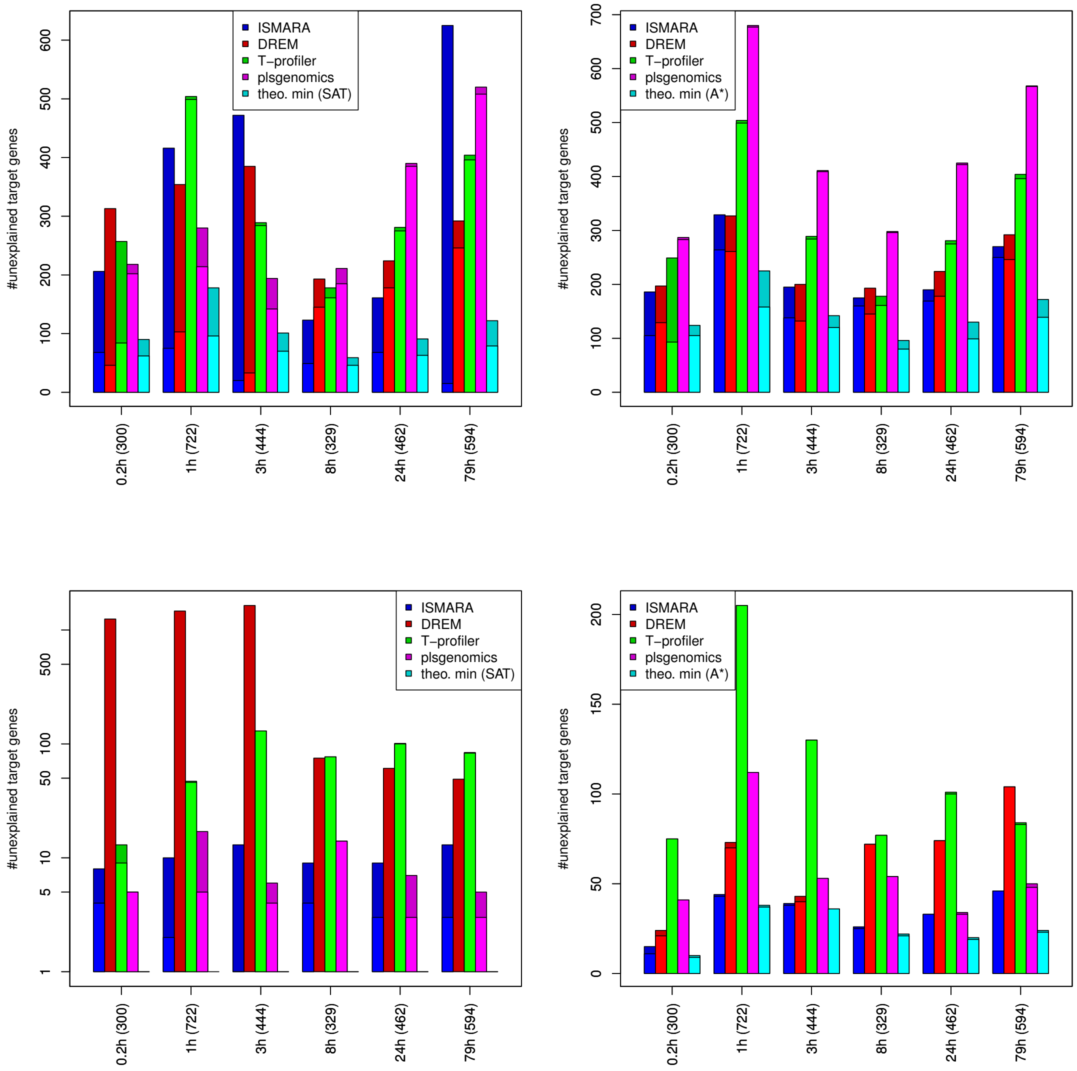


Figure 2.3: The number of unexplained target genes (#UTG) for the respiratory shift from 20.9%-oxygen time series and the YEASTRACT (top) and ISMARA (bottom) network. On the left all predicted TFs are used and on the right the 10 most important TFs only. The brighter part of the bars indicates how many of the UTGs have changed significantly in the data. The darker part of the bars correspond to the UTGs which are unchanged. For each condition the number of differential genes is given in parentheses.

2.3.2 Performance of methods depends on the particular experiment

To assess the different predictions, we calculated the *i-score* and #UTG for the predictions of all methods for the different datasets for both the YEASTRACT and the ISMARA networks. For details on how we derived activity changes from the predictions see the Supplement. The #UTG for the respiratory shift data and the YEASTRACT network are shown in the upper part of Figure 2.3. For each condition in the dataset the #UTG is given for the different methods. On the left this is shown for the complete prediction while the right plot shows the prediction restricted to the top 10 changing TFs. We also provide the #UTG if the *i-score* is optimized directly, for the unrestricted case on the left by the Act-SAT optimization and on the right by the Act-A* method. The brighter part of the bars indicates how many of the UTGs were differentially expressed.

All methods yielded many UTGs for which the observed effect could not be explained. According to the #UTG, there is no clear best performing method, as these numbers vary considerably across conditions. For the first three timepoints *DREM* and *ISMARA* appear to predict too many active TFs with many targets, as most of the unexplained genes were not significant and there are fewer unexplained significant genes compared to the other methods. This also explains why the scores improve if only the 10 most changing TFs are considered. Only for *plsgenomics* there are considerably more UTGs if only the 10 most changing TFs are considered. This indicates that *plsgenomics* predicts many TFs each of which explains only a small portion of the observed effects. The optimal solutions with respect to the *i-score* for an unlimited number of active TFs (calculated by Act-SAT) and for the 10 most changing TFs (Act-A*) are also comparable so that it appears that 10 active TFs are sufficient to explain the majority of the observed effects for this dataset.

Surprisingly, for the respiratory shift data and the YEASTRACT network there are about 100 target genes that are unexplained even if the *i-score* is optimized. As we make only minimal assumptions and, thus, underestimate the #UTG this is a surprisingly large number. The optimization inherently assumes that the network is true and complete, but current gene regulatory networks contain condition specific edges and are incomplete. If a network contains many incorrect edges solutions yield a good score because these wrong edges are used to explain the effects. As additional edges can only improve the score, UTGs have to be caused by noise in the data or missing edges in the network.

Using the more dense ISMARA network for the prediction and calculation of the *i-score*, the results for the different methods show a larger variation (see bottom row in Figure 2.3). Especially *DREM* does not seem to be well adapted for such a dense network and predicts too many active TFs. Thus, a huge number of unchanged genes are unexplained as all their associated TFs are predicted to be active and all changing genes are explained. Again, if only the top 10 changing TFs are considered *DREM* performs comparable to the other methods. *T-profiler* also yields many UTGs, but in contrast to *DREM* there are even more UTGs if only the top 10 changing TFs are considered. Almost all UTGs of *T-profiler* were significantly changed in the expression data. Possibly, in the dense ISMARA network the TFs that really regulate these genes are also associated with many unchanged genes so that the t-test is insignificant and the TFs are not predicted to be active. In general, the two regression-based methods *ISMARA* and *plsgenomics* clearly perform better than *DREM* and *T-profiler* if the dense ISMARA network is used.

Moreover, we analyzed the number of unexplained target genes using the alternative definition of unexplained target genes that takes complementary TF configurations into

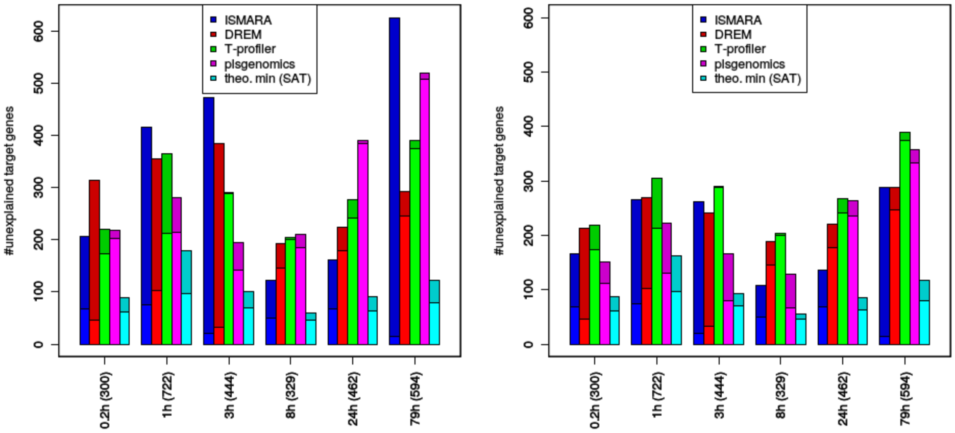


Figure 2.4: Comparison of the number of unexplained target genes (#UTG) with the normal definition of unexplained genes as used in the paper (left) and the alternative definition that also allows unchanged target genes to be explained if all its TFs are changing (right).

account. According to this definition an unchanged target gene can only be unexplained if all its TFs have the same regulatory effect. Figure 2.4 shows the #UTG for both definitions of unexplained target genes. For plsgenomics we optimize the activity threshold above which a TF is active, so that the set of active TFs is different for the two UTG definitions. The number of unchanged unexplained target genes decreases especially for ISMARA and DREM, but there are still many unexplained target genes for all methods.

Overall, when the ISMARA network is used fewer unexplained target genes are observed. However, this does not necessarily mean that the predictions are closer to the truth. The ISMARA network (155.404 edges) is much denser than the YEASTRACT network (41.498 edges). So, the genes are associated with more TFs and it is more likely that at least one activity change is predicted which yields a consistent edge.

2.3.3 Assessment of networks

To compare the different networks with respect to the *i-score*, we compared the optimal score determined by Act-A* for real and randomized data. For this we shuffled the genes of the Gasch compendium [37] 100 times and for each such random dataset calculated the optimal Act-A* solution for all 173 conditions. For each of the conditions z-scores comparing the *i-score* of the real data compared to the 100 randomized runs are calculated for all networks. Furthermore, we calculated z-scores in the same way for random networks with the same number of edges. To generate the random networks, we kept the TFs and target genes from the original network and added as many random edges as were in the original.

Figure 2.5 shows the z-score distributions for the YEASTRACT, ISMARA, Jaspar and

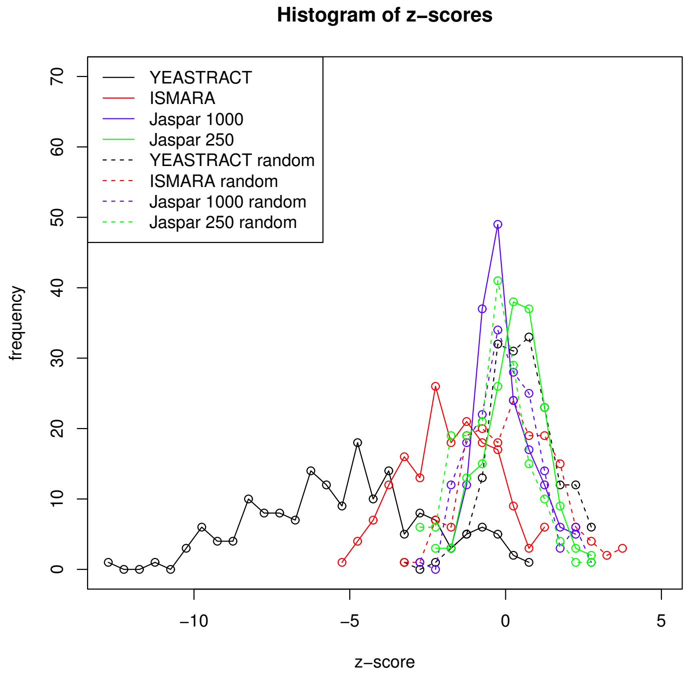


Figure 2.5: Z-score distributions of the comparison of the *i-scores* of the Act-A* solution for the real and randomized Gasch data. A negative z-score indicates that the *i-score* was smaller/better for the real data than for the randomized data. These z-scores were calculated for the YEASTRACT, ISMARA and Jaspar network and a random network with the same number of edges for each of these networks. When the YEASTRACT network is used the *i-score* is much better for the real than for the random data, whereas the scores are about the same for the Jaspar network.

the corresponding random networks. A negative z-score indicates that the *i-score* of the real data was smaller than for the randomized data. For the ISMARA and YEASTRACT networks the z-scores for most conditions are negative, while for the Jaspar network and the randomized networks the distribution of the z-scores is centered at 0. For randomized networks there are about equally many unexplained target genes for both the real and randomized data. For both YEASTRACT and ISMARA there is a clear distinction between the z-score distribution for the real and the random network. The network constructed by the Jaspar binding site motifs performs not better than the corresponding random network. The *i-scores* that are calculated using the YEASTRACT network can discriminate better between real and random data as compared to the ISMARA network.

2.3.4 Variability of solutions

To investigate how much solutions with a good *i-score* differ from each other, we analyzed the best 10% of all solutions scored during the Act-A* optimization. Figure 2.6 shows boxplots (top) of the obtained *i-scores* improvements and the corresponding solu-

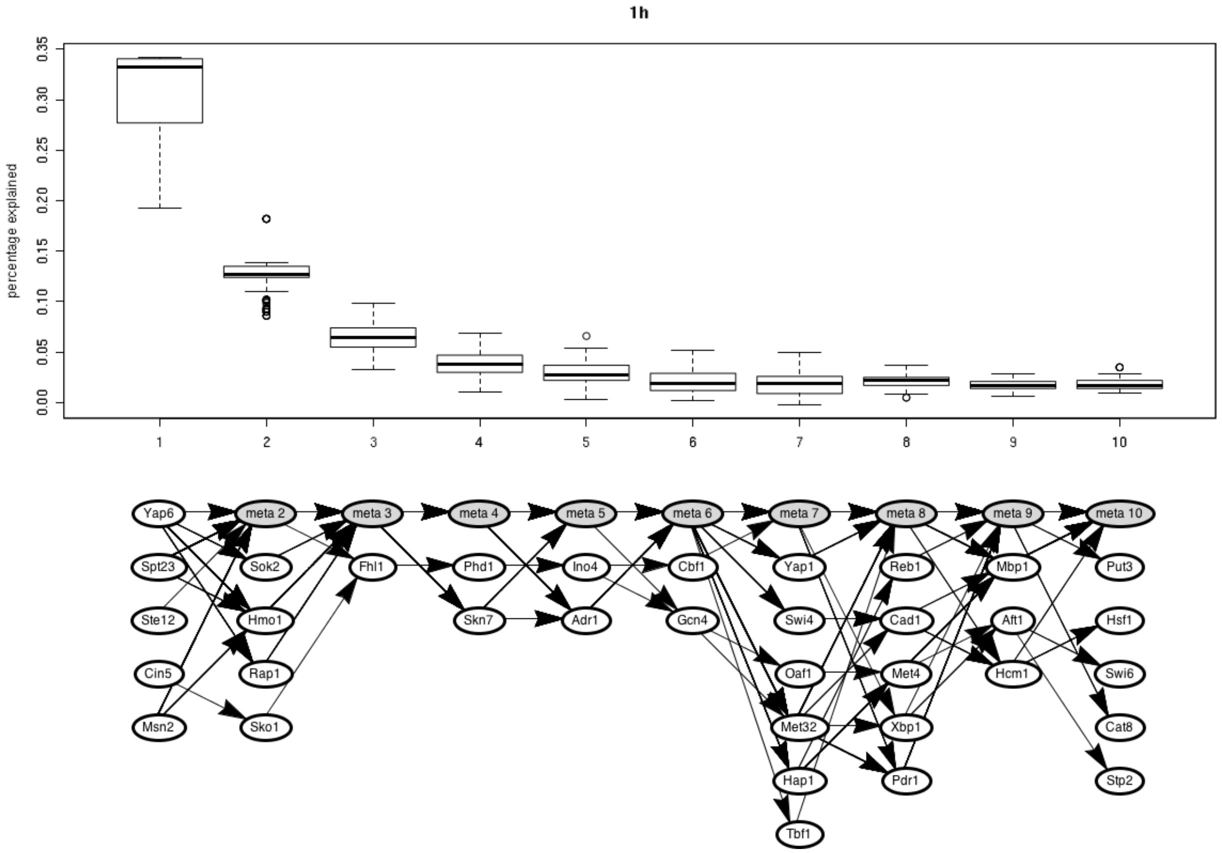


Figure 2.6: Variability of the 10% best Act-A* solutions. The graph (bottom) shows the active TFs in these solutions. Each path in the graph corresponds to one solution. The position indicates the A* step in which the respective TF is added to the solution. TFs that were added in another solution at an earlier position are collapsed into meta nodes. The boxplot above shows the relative score improvements of the TFs at the given position. Only the 14 shown TFs are used in the first five positions to explain the majority of the effects.

tions (bottom). Every solution corresponds to a path in the graph. The position of the active TFs in the path indicates at which step of A* it was added to the solution. If a TF is included in different solutions at different positions only the first is shown and for the other solutions a meta node is introduced at the corresponding position. These meta nodes contain all TFs that were present at an earlier position during the optimization in other solutions. This way, each TF is only present once in the graph at its first position. The boxplot above the graph shows the relative improvement of the *i-score* caused by the addition of the TFs at this position.

The first 5 TFs in any solution explain most of the effects, while the other TFs only explain small fractions of the unexplained target genes (each about 2.5% of the *i-score*). Moreover, there are relatively few alternative TFs used at the first positions, all solutions used some combination of 14 different TFs for the first 5 positions. For the larger positions there are more alternatives that all yield approximately the same (very small) score contribution. So, while the most important TFs are relatively robust across well scoring solutions, the less important TFs are more variable.

2.3.5 Application to human data

The human gene regulatory network is much larger and more complex than the yeast regulatory network. [44] To demonstrate and evaluate our approach on human data, we constructed a context-specific network from DNase I hypersensitivity and ChIPseq data for two ENCODE cell lines [26] and experimentally validated miRNA-target regulations [52]. The resulting network was used to find the active TFs which yield the minimal *i-score* for the RNAseq data of the corresponding cell lines. Even though the gene regulation is more complex in human and the network might be of a poorer quality, only 261 target genes were inconsistent for 20 active TFs, and many of the active TFs were biologically meaningful (see Supplement for more information).

2.4 Discussion

The prediction of differentially active TFs is an important task for which several tools are available whose performance could so far not be compared systematically. We have shown that the predictions of different methods differ greatly, so that it strongly depends on the used method which TFs are predicted to be differentially active, especially if only the most important TFs are analyzed.

We propose an evaluation strategy to assess how many of the measurements are unexplained for a given set of differentially active TFs. We make minimal assumptions so that we only define a gene to be unexplained if there cannot be a reasonable activation function for the associated TFs such that their activities fit to the measurement. The real activation function, however, is unknown so that it is possible that genes that we assume to be explained are actually unexplained, as the true activation function applied to the predicted active TFs does not result in the observed effect. Thus, the real unexplainability is (grossly) underestimated.

The *i-score* is easy to use and interpret. The number of unexplained target genes (#UTG) is straightforward and gives, especially together with the theoretical minimum calculated by the Act-SAT or Act-A* method, an intuitive measure of how well the prediction fits to the data. To use the *i-score* only a list of TF activity changes is needed as input (in addition to the data and network that were used for the prediction).

The comparison of the different methods showed that the results strongly depend on the condition as well as the used network so that we could not identify a clear best method. Our analysis did show that not all methods are equally suited for all networks, as some methods are designed to use high quality experimentally derived networks and other methods for more dense (often only inferred) networks. The *i-score* can help to decide which method is best suited for the given network.

Surprisingly, in the evaluation using the YEASTRACT network (which is the current gold standard of yeast gene regulatory networks) many genes were unexplained even for the directly optimized theoretical minimum calculated by Act-SAT or Act-A*. This means that for many genes it is *not* possible to explain the observed effects with the current networks likely because of missing edges in the network. As we make only minimal assumptions and, thus, underestimate the *i-score*, the actual number of unexplained cases will be even higher.

The Act-A* optimization provides the possibility to include prior knowledge. If some TFs are known to be active in the analyzed condition, the Act-A* optimization can be started from the partial solution in which these TFs are set active and then find other active TFs that fit best to the not yet explained effects.

Furthermore, the *i-score* can also be used to explore the effect of individual TFs in a given prediction, by comparing the scores of the solutions where this TF is set to the more (A^+)/less (A^-) active state and inactive (A^0) state, respectively. This way it can be determined whether there is an alternative solution with similar score which does not use the TF in question. Moreover, it allows to add new edges (potential new regulations) or to remove edges and to compute the difference in the i-score. Thus, new regulatory hypotheses can be assessed in the context of the current regulatory network and for the observed data at hand.

2.5 Conclusion

The results of the prediction of differentially active TFs differ greatly between methods and so far there are no systematic approaches and associated evaluation criteria that can be used to assess the performance of different methods. In this study we propose the inconsistency score that evaluates whether given activity changes can explain the measured expression changes. Furthermore, we propose two optimization approaches to determine the theoretical minimum of this score given the data and network. Together, the theoretical optimum and the score for a given prediction are good measures to assess the reliability of the activity changes of TFs and the theoretical optimum can be used to evaluate different networks and to evaluate regulatory hypotheses. Thus, the *i-score* is a useful tool for the analysis of any large-scale dataset.

Chapter 3

Comparison of Six Breast Cancer Classifiers using qPCR

Motivation

Not for all applications a completely independent evaluation measure such as the *i-score* is possible. In many applications there is one 'gold standard' evaluation measure that is used to evaluate all methods. To nevertheless ensure that the methods do not overfit, independent test sets are used.

One such application is the evaluation of breast cancer subtype classifiers. Studies usually use an independent cohort with survival data for evaluation, but they typically do not compare themselves against other classifiers. As the different classifier do not use the same independent cohort for validation it is hard to evaluate their performance. There are some meta-studies available that compare several classifiers on the same cohort, but they mostly use microarray studies, even though many available classifiers are based on qPCR measurements.

We used a prospective study of 726 early patients from five certified German breast centers that were treated according to national guidelines and for which the gene expression of 94 genes have been measured by the mid-throughput qPCR platform Fluidigm. Clinical and pathological data as well as information on outcome over five years is available. Using this data, we compared the performance of six classifiers: scmgene and research versions of PAM50, ROR-S, recurrence score, EndoPredict and GGI.

Overall, we found a high concordance between most of the classifiers and also a high prognostic performance. The classifiers that were originally developed for microarray data still performed well using the Fluidigm data. Therefore, Fluidigm can be used to measure genes of several classifiers. Moreover, their results can be compared for an improved prognosis.

In addition, we provide an interactive report of the results, which allows analysis of differences between the classifiers down to the individual patients and their characteristics. This not only makes our results more transparent, but also allows an in-depth analysis and comparison of the classifiers.

Publication

The content of this chapter was submitted to Bioinformatics ([11]). Here, the introduction was rewritten to given a more general background of breast cancer and subtype classification.

Author Contributions

Martina Vetter and Eva Kantelhardt designed and executed the study. Christine Fathke edited the manuscript. Christoph Thomsson supervised the study. Melanie Gündert performed the Fluidigm measurements and Susanne Ulbrich supervised the measurements. Evi Berchtold analyzed the data and wrote the manuscript. Evi Berchtold and Gergely Csaba designed the methods. Ralf Zimmer supervised the analysis and edited the manuscript.

Availability

The interactive report of the results is available at

`https://services.bio.ifi.lmu.de/pia/`

3.1 Introduction

3.1.1 Cancer

Cancer is a disease where cells divide uncontrolled and spread to the surrounding tissues and form metastases. It can occur in nearly any tissue and even within a tissue the disease is diverse with several subtypes. In 2000 Hanahan and Weinberg [48] defined six hallmarks of cancer, that is abilities that each cancer needs to develop to transform into malignant cells.

Normal cells only proliferate when mitogenic growth signals are detected by their transmembrane receptors. Cancer cells limit their dependence from these external signals by producing their own growth signals, increasing the growth signal receptors such that a lower concentration of growth signals suffices to trigger proliferation or alter the downstream signal cascade. Similarly, cancer cells need to be insensitive to antigrowth signals that block proliferation and programmed cell death (apoptosis). Again, this can be achieved by downregulating or disrupting the receptor, or by altering the downstream signal cascade.

Normal cells can only replicate a limited number of times as the telomeres at the end of the chromosomes are shortened by each replication. When the telomeres get too short, the ends of the chromosomes are no longer protected which leads to end-to-end chromosomal fusions, which results in cell death. The enzyme telomerase is able to add additional hexanucleotide repeats to the telomeres. Almost all cancer cells upregulate the expression of telomerase to keep their telomeres long enough.

As the tumor grows, new blood vessels that supply nutrients and oxygen are needed. Only when the tumor cells gain the ability to encourage blood vessel growth, the tumor can keep on growing. Angiogenesis is activated by changing the balance of angiogenesis inducers and inhibitors, either on the gene expression level or directly on the protein level.

When space and nutrients become scarce tumor cells move to adjacent tissues and form metastases. For this, cell-to-cell interaction mechanisms are altered and extracellular proteases are upregulated.

In 2011 Hanahan and Weinberg [49] updated their list of hallmark abilities, by two emerging hallmarks and two enabling characteristics. Cancer cells seem to change their energy metabolism. If oxygen is available normal cells process glucose first via glycolysis to pyruvate and subsequently to carbon dioxide in the mitochondria. Under anaerobic conditions only little pyruvate is processed in the mitochondria. Cancer cells, however shift their energy metabolism to a state called aerobic glycolysis, where only little pyruvate is processed by the mitochondria even though oxygen is available. One possible explanation of this phenomenon is that glycolysis provides many intermediates that can be used to build e.g. amino acids and nucleotides which are needed in the proliferating state of cancer cells.

The other emerging hallmark is that cancer cells have to evade destruction by the immune system. The two enabling characteristics are tumor-promoting inflammation and genome instability. Both these characteristics help the tumor cells to acquire the hallmark abilities. Inflammation can supply bioactive molecules to the tumor including growth factors or survival factors. Genome instability can change the epigenome as well as increase the mutation rate, both of which facilitate the acquisition of hallmark abilities.

All these hallmark abilities can be acquired by different mechanisms and in different order. Some mutations can even result in multiple hallmark abilities.

3.1.2 Breast cancer subtypes

Breast cancer is the most prevalent cancer. Like most cancers, it is not a homogeneous disease, but consists of multiple subtypes. Hierarchical clustering of breast cancer gene expression measurements yields five intrinsic subtypes: Luminal A/B, HER2 overexpression, basal and normal-like[23]. Table 3.1 shows an overview of these subtypes. Interestingly, the subtypes defined by gene expression correspond to subtypes defined by a few immunohistocheminal (IHC) markers, only the Luminal A and normal-like subtypes have the same IHC marker status.

The different subtypes differ in their prognosis as well as in their treatment choices. Luminal tumors express the hormone receptors ER and PgR that transfer proliferation when bound by the corresponding hormone. Among these luminal tumors there are at least two subtypes (Luminal A and B) that approximately differ in their HER2 status. Luminal A tumors are HER2 negative and have good prognosis. This subtype is typically treated with hormone therapy and responds poorly to chemotherapy. Luminal B tumors overexpress the HER2 growth factor receptor, have a worse prognosis compared to Luminal A tumors and are treated by a combination of hormone treatment and chemotherapy.

Intrinsic subtype	IHC status	Grade	Outcome	Prevalence
Luminal A	[ER+|PgR+]HER2-KI67-	1|2	Good	23.7%
Luminal B	[ER+|PgR+]HER2-KI67+	2|3	Intermediate	38.8%
	[ER+|PgR+]HER2+KI67+		Poor	14%
HER2+	[ER-PgR-]HER2+	2|3	Poor	38.8%
Basal	[ER-PgR-]HER2-,basal marker+	3	Poor	12.3%
Normal-like	[ER+|PgR+]HER2-KI67-	1|2|3	Intermediate	7.8%

Table 3.1: Intrinsic breast cancer subtypes, with IHC marker status, grade, outcome and prevalence. All subtypes differ in their IHC status, except Luminal A and Normal-like. Data taken from Dai et al.[23]

HER2 overexpressing tumors are negative for the ER and PgR and overexpress the HER2 growth factor receptor. The prognosis for this subtype is poor and it is typically treated by a HER2 antibody (trastuzumab) and chemotherapy. Basal tumors are triple negative (ER-,PgR-,HER2-) and their expression profile is similar to basal epithelial cells. The prognosis is poor and there is no target therapy for basal tumors so that chemotherapy is the only option.

3.1.3 Subtype classifiers and risk scores

As the correct classification of the tumor subtype is important for the choice of treatment, many different methods were developed to predict the subtype. In principle, there are two different approaches: classifiers that predict the subtype for a given tumor, and risk scores that predict the risk of recurrence (often for a given IHC subtype).

PAM50 [77] is a subtype classifier that uses the expression of 50 genes to predict the intrinsic subtype of a tumor. To derive this classifier they used a list of 1906 intrinsic genes to cluster 189 breast cancers. This clustering yielded 5 clusters that corresponded to the intrinsic subtypes and overall covered 122 of the 189 breast cancer samples. The list of intrinsic genes was filtered for qRT-PCR quality and sorted by their t-test statistic between the clusters. The top 50 genes were selected for the classifier. To predict the subtype of a new sample, the Prediction Analysis of Microarray (PAM) method [94] was used. Furthermore, two risk scores that use the subtypes were derived using a multivariable Cox regression considering only the correlation to the subtypes (ROR-S) or also including the tumor size (ROR-C).

The scmgene [46] subtype classifier is not derived from hierarchical clustering, but uses a combination of three Gaussian distributions. The three Gaussian distributions correspond to an ER, an HER2 and a proliferation module. The modules can consist of several genes that are related to the process, but in case of scmgene they consist of single genes: ER, HER2 and AURKA for the proliferation module. Three clusters along the ER and HER2 modules are identified and represented by a Gaussian distribution. A new patient is assigned to the subtype of the Gaussian with the highest posterior probability. If the patient is assigned to the ER+/HER2- subtype, the proliferation module is used to

risk score	#genes	#patients train	#patients test	subtype
GGI	8	77	139; 270	ER+
EndoPredict	11	964	378; 1324	ER+,HER2-
recurrence score	21	447	668	ER+
ROR-S	50	189	761	-

Table 3.2: Overview of risk scores. For each risk score the number of genes used to calculate the risk score, the number of patients in the trainings and test sets, and the IHC subtype for which it is used is given.

decide whether the patient is ER+/HER2- High Prolif. or ER+/HER2- Low Prolif. They compared the performance of scmgene to other classifiers and risk scores and found that it was more robust across different patient cohorts.

Table 3.1 gives an overview over some risk scores. Risk scores are typically trained for a given IHC subtype, e.g. ER positive patients. They use comparatively few genes that can easily be measured by qPCR. To calculate the risk score, in most cases a weighted sum of the expression values is calculated and a predefined threshold indicates whether the patient is at high or low risk of recurrence.

Several of these signatures have been developed to commercial assays and are now also used in clinical practice. In the last years, there were two large prospective randomized trials that analysed the survival of patients who received treatment according to the classification of Mammaprint (70 genes, [19]) and the recurrence score (21 genes, [89]).

In 2011, Venet et al. [96] reported that gene sets that are completely unrelated to breast cancer or even random gene sets can yield significant p-values for the prediction of risk of recurrence for breast cancer patients. Given this observation it seems hazardous to simply report a significant p-value on some cohort when presenting a new classifier, as is routinely done. Instead, the new classifier should be compared to existing classifiers to show that it has some advantage, e.g. improved performance, robustness or applicability. Furthermore, the already published classifiers need to be evaluated systematically on independent test sets that were not used in the development of any classifiers. In the last years, a few such studies have been published [30, 47, 78, 71], but as a comparison of several classifiers requires a large number of measured genes, all these studies used microarray measurements, even though many of the available classifiers have been developed for qPCR measurements of the gene expression.

3.1.4 Fluidigm Dynamic Array IFC

The Fluidigm Dynamic Array IFC qPCR platform [90] can help to decrease the cost of measuring the gene expression of many genes, as needed for breast cancer classifiers. For most classifiers the gene expression of several genes is measured by qPCR. Traditional qPCR platforms require that each combination of patient sample and primers of the genes are pipetted together individually to be measured. This results in $patients*genes*2$ pipetting steps. The Fluidigm IFC platform has a system of fluid lines and valves that automatically distribute the RNA samples and primers to the individual reaction chambers without mixing them. So only *patients+genes* pipetting steps are needed to measure hundreds of genes for hundreds of patients.

3.1.5 Prognosis in everyday routine (PiA) study

We have used the Fluidigm IFC platform to measure the expression of 94 genes for a large cohort of 726 patients. We selected the 94 genes such that they cover six different breast cancer signatures: PAM50 and the corresponding risk score ROR-S [77], scmgene [46], EndoPredict [34], Genomic Grade Index (GGI) [33] and the recurrence score [76]. For all classifiers the research versions were used. Thus, we can compare the prognostic power of these signatures on an independent routine cohort on which none of the signatures was trained and provide a first study that compares the performance of breast cancer signatures on qPCR data obtained in a standardized manner.

3.2 Methods

3.2.1 PiA cohort

Within the multicenter prospective PiA trial (NCT 01592825) tumor tissue samples of consecutively diagnosed breast cancer patients from five German certified breast centers were collected at Martin-Luther University, Halle-Wittenberg between 2009 and 2011. Female patients with operable, non-metastasized breast cancer independent of lymph node status were included. The study was approved by the ethics committee of the Martin-Luther University Halle-Wittenberg and each patient gave informed consent. A total of 726 fresh frozen samples of primary tumor tissue were investigated using Fluidigm IFC platform [90]. Tumor specimens were fresh frozen after surgery and stored at -80 °C until further use. Tumor content was verified histologically. Clinical and pathological parameters were obtained for each patient and documented using SPSS 24 (SPSS Inc., Chicago, Illinois, USA). TNM staging system was used [87]. Information on therapy applied was not available. Patient information was anonymized prior to analysis. Receptor defined breast cancer subtypes were determined according to the St. Gallen classification [43]. Due to missing Ki-67 values, we used histopathological grading to assess cell proliferation [99]. The following system was applied to define histopathological subtypes:

- Luminal A-like: Estrogen receptor (ER) positive, Progesterone receptor (PgR) positive, HER2 negative, grade 1 or 2.

- Luminal B-like (HER2 negative): ER positive, PgR negative, HER2 negative or grade 3.

- Luminal B-like (HER2 positive): ER positive, HER2 positive, any grades.

- HER2 positive (non-luminal-like): ER negative, PgR negative, HER2 positive, any grade.

- Triple negative breast cancer (TNBC, Basal-like): ER negative, PgR negative, HER2 negative, any grade.

Expression of 94 genes was measured using the Fluidigm qPCR platform. This amounts to 726 x 94 = 68.244 qPCR reactions. To ensure that the measurements of the Fluidigm platform are of good quality and comparable across chips, for all samples five genes were also measured on the CFX384 qPCR platform, so that the results could be compared. This platform uses 384 well plates, so that qPCR measurements for one gene can be done in parallel for 384 samples.

An overview of the clinical characteristics of the patients and tumors are shown in Table 3.3. Most of the patients (610 of 726) are ER positive and only a small subset (104) is HER2 positive. The majority of the tumors had histological grade 2 and lymph nodes were not affected.

The standardized definitions for efficacy end points (STEEP) criteria were used as endpoint definitions [54]. The primary endpoint of this study was overall survival (OS). Person time equaled the time from the date of diagnosis to the date of event or to the date of last contact. Women without event were right-censored at the last visit to the clinic.

3.2.2 Normalisation

On one Fluidigm IFC chip 96 genes can be measured by qPCR for 96 samples. Thus, the 726 patients have been measured on several chips that need to be normalized to make them comparable. There are three sources of bias when several Fluidigm chips are measured: the amount of cDNA can differ between samples (within a chip and between chips), there can be variation between the chips, e.g. due to different efficiency of the PCR reactions and there can be differences in the pre-amplification of the cDNA that is necessary for the Fluidigm platform. To correct for variation between chips, so called inter plate calibrator (IPC) samples, are measured on each chip. The difference between cDNA amounts of individual samples can be diminished, by using the expression of genes that are expected to be constant between samples, e.g. housekeeping genes. Most classifiers already include housekeeping genes for normalization purposes so that no additional genes have to be measured. The cDNA has to be pre-amplified before it is loaded on the Fluidigm IFC chip. Amplification for all 96 primers at once can generate problems, so that we splitted the set of primers in two subsets that are amplified individually. For this we tried several different batches and used the division that yielded most successful amplifications. However, there can be differences between the efficiencies of the pre-amplification reactions. This can be corrected as one can assume that the median of all measurements of each chip and pre-amplification mix is the same. For more information on the individual normalization steps see the Supplement.

	all	**Luminal A-like**	**Luminal B-like (HER2 negative)**	**Luminal B-like (HER2 positive)**	**HER2 positive (non-luminal-like)**	**Triple negative breast cancer (TNBC, Basal-like)**	**not classified**
#patients	726	378	163	69	34	74	8
grade							
1	76	67	4	3	0	0	2
2	447	311	59	40	12	22	3
3	203	0	100	26	22	52	3
size							
<1	42	22	9	2	4	5	0
1-2	302	176	69	24	13	16	4
2-5	341	161	77	37	16	46	4
>5	41	19	8	6	1	7	0
nodal status							
0	450	239	102	41	21	42	5
1	201	108	48	16	8	20	1
2	47	22	5	7	4	7	2
3	28	9	8	5	1	5	0
age							
avrg	62.62	62.46	64.89	59.19	61.32	63.11	54.25
min	22	22	29	28	31	25	30
max	90	89	90	86	81	88	75
survival							
alive	630	348	136	58	28	53	7
deceased	96	30	27	11	6	21	1

Table 3.3: Clinical characteristics of the PiA cohort, grouped by histopathological subtype. Patients that do not fall in any category described in 3.2.1 are shown in the last column.

3.2.3 Classification

The genefu R package [40, 80] was used to calculate the PAM50, scmgene, ROR-S and recurrence score. The PAM50 classifier can be applied in two ways: the published centroids can be used directly for the prediction, or the centroids are first trained on the given dataset and then used to predict the subtypes. As a high C(t) value indicates low gene expression whereas a high microarray intensity indicates high gene expression, the C(t) values were not used directly for these microarray based methods, instead the difference to the maximal PCR cycle $C(t)_{max}$ was used. For GGI and EndoPredict the formulas from the corresponding papers were re-implemented and the published cutoffs were used for EndoPredict. For GGI no published cutoff is available, so that we used the median to divide the cohort in two equally sized groups. All classifiers are applied to the complete cohort.

3.2.4 Performance and Concordance of Predictions

To assess the performance of the predictions, we generated Kaplan-Meier plots and calculated the concordance index (c-index) for each classifier. The c-index corresponds to the probability that for a pair of randomly chosen samples, the sample with the higher risk score experiences an event before the other sample.

As we are able to calculate several classifiers for the same cohort, we compared their predictions by calculating Cramer's V. This statistical measure quantifies the correlation between two predictions. It ranges between 0 and 1, with values above 0.5 indicating a strong association. We compared subtype classifiers (PAM50 and scmgene) and risk scores separately, to account for the different number of predicted groups.

Moreover, we used multivariate Cox regression to create a combined predictor that uses the risk scores of the different classifiers as input. For this, only risk scores that return a numeric risk score were used (excluding PAM50 and scmgene) and their scores were scaled, so that scores yielding a low risk prediction (i.e. having a score below the corresponding cutoff) are mapped to 0-0.5 and high risk scores to 0.5-1. Most risk scores are not able to return a score if one of the measurements is missing due to technical errors during the measurement. In this case, the combined risk score is also not able to return a score. As this is more probable when more genes are used, the combined risk score cannot return a score for many patients. To nevertheless return a score for these patients, we trained multiple models, excluding each risk score in turn. For the final prediction we used the model that uses all risk scores, and only used one of the restricted models if the complete model does not return a risk score. To evaluate the performance of this combined risk score, a five-fold cross validation was used to prevent overfitting.

3.2.5 Robustness of Classifications

Like all measurements, also gene expression measurements are subject to noise. As most subtype classifiers use a combination of many genes, the impact of noisy measurements is reduced, as no single gene influences the prediction too strongly. To assess the impact of noise on the prediction, we simulated noisy measurements and checked how often the prediction changed due to small changes in the gene expression data. For this we repeatedly sampled for each measurement a noise term from a normal distribution centered around zero and added it to the measurement. Then we checked for each classifier, whether the same subtype or risk group (high or low) was predicted for the real and modified measurement. Robust classifiers should be able to make the same prediction for the real and modified measurements with simulated noise in most cases.

A similar approach allows us to estimate the probability that a single noisy measurement results in a false prediction for a given patient. For this we calculate for each gene contained in the classifier the minimal difference of the gene expression value that would result in a different prediction. For classifiers with simple formulas this can be calculated directly, while it can be sampled by calculating the score with a growing noise term for more complex classifiers. Given a background noise distribution (e.g. a normal distribution with mean zero) the probability of observing at least as much noise can be calculated. These probability values can help to identify gene expression measurements for which already small (i.e. highly probable) deviations have an effect on the prediction. For these measurements replicate measurements can then be considered to reduce the impact of random noise and improve the quality of the prediction.

3.2.6 Interactive Report

In addition to the results presented in this paper, we provide a website that contains an interactive report of the results (https://services.bio.ifi.lmu.de/pia). The overview page contains all the main results: the clinical and pathological characteristics table, performance table, coherence plot and Cramer's V table and additionally an overview of all features for all patients. In the clinical characteristics table for large enough patient groups with similar characteristics the performance results for this subcohort can be analyzed. Moreover, for each entry in the performance table the corresponding Kaplan-Meier plot can be shown in a popup window, to evaluate the performance in more detail. The survival endpoint used in the Kaplan-Meier plot can be selected to directly compare the influence of the different survival endpoints. Furthermore, a page comparing two classifiers is linked to the corresponding entry of the Cramer's V table. This comparison page shows both Kaplan-Meier plots side by side, so that they can be compared directly. Furthermore, a contingency table shows how many patients are classified with a given combination of classifications of the two selected classifiers. This table is again linked to a list of the corresponding patients, with all available clinical features, classifications and survival information. This way, one can analyze the patients that were classified discordantly in full detail. The patient overview table is linked to a details view for each individual patient. This view not only shows the available features of this patient, but also for each classifier an overview of the corresponding gene expression measurements and how they relate to the distribution of the gene expression measurements of the whole cohort, or the subsets that experienced an event or not. Furthermore, the minimal difference in gene expression to change the prediction and the corresponding probability to experience this difference due to random noise is shown for each gene contained in the classifier. Such a detailed view on individual patients can greatly help to understand individual predictions and the influence of the contained genes.

3.3 Results

3.3.1 Comparability of Fluidigm Chips

With appropriate normalization the different Fluidigm chips should be comparable. To test this, the CFX and Fluidigm measurements were compared for the five genes that were also measured on the CFX platform. Figure 3.1 shows the comparison of the C(t) values of the two platforms for the reference gene RPLP0. The different Fluidigm chips are highlighted

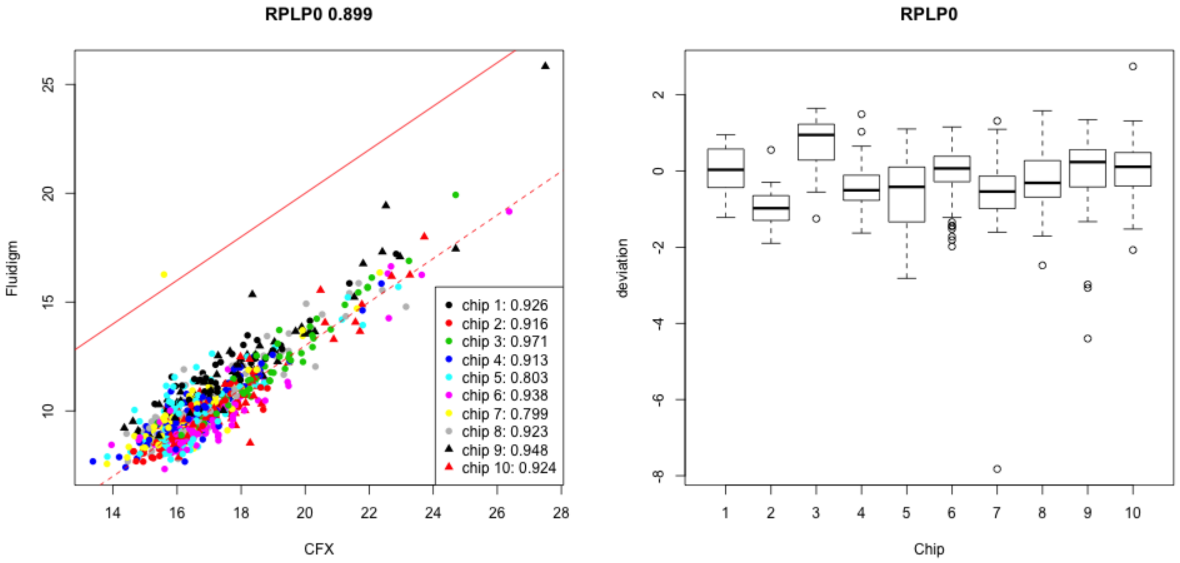


Figure 3.1: Comparison of C(t) values for the RPLP0 gene for 726 samples measured on 10 Fluidigm chips and the CFX platform. On the left the C(t) values are scattered against each other. The overall correlation as well as the correlations for each Fluidigm chip are given in the title and legend of the plot. There is a shift in the absolute C(t) values due to different cDNA concentrations and the pre-amplification, but there is a clear correlation between the two measurements and no apparent bias between the Fluidigm chips. The plot on the right shows the deviations between the Fluidigm and CFX measurements for each Fluidigm chip separately.

by different colors and there is only some bias for chips 1 and 2. For the first three chips the sample amounts differed slightly as they were not done in one batch with the other chips. This variation is normally corrected for by the housekeeping normalization that was not applied for this comparison due to the small number of genes on the CFX platform. The concordance between the two measurements is quite good with only few outliers. The C(t) values are shifted between the two qPCR platforms as they are using different amounts of cDNA and the cDNA is pre-amplified for the Fluidigm platform. But in general, the two platforms agree very well, so that the Fluidigm platform seems to be suitable for its use in gene expression profiling also of large cohorts using multiple chips.

3.3.2 Survival Analysis

For the PiA study five year survival data is available for which we analysed the overall survival (OS), invasive disease-free survival (IDFS), distant disease-free survival (DDFS) and recurrence-free intervall (RFI), all defined according to STEEP criteria [54]. In this paper we focus on overall survival, the results for the other endpoints can be found in the interactive report. The survival data was used to calculate different measures for the performance of the risk scores: hazard ratios, logrank p-values and the concordance index

	OS				
risk score	**logrank p**	**HR**	**c-index**	**# event**	**# no event**
recurrence score	8.29e-5	4.49	0.70 (0.49-0.85)	19/8	74/151
EndoPredict	4.089e-6	3.77	0.69 (0.57-0.78)	79/12	365/230
EPclin	1.12e-6	3.42	0.72 (0.61-0.81)	74/17	320/275
GGI	9.19e-6	2.61	0.64 (0.52-0.73)	68/28	295/335
ROR-S	3.03e-6	3.43	0.68 (0.57-0.77)	88/8	430/200
combination	1.09e-7	4.07	0.72 (0.61-0.81)	82/9	367/230
PAM50	1.678e-5	3.82	-	72/24	331/299
scmgene	0.001	1.48	-	53/12	313/183

	RFI				
risk score	**logrank p**	**HR**	**c-index**	**# event**	**# no event**
recurrence score	6.512e-3	4.36	0.70 (0.42-0.89)	10/4	83/155
EndoPredict	3.229e-7	11.06	0.78 (0.64-0.87)	58/3	386/239
EPclin	7.159e-8	7.23	0.80 (0.67-0.89)	55/6	339/286
GGI	2.871e-7	4.29	0.70 (0.57-0.81)	53/13	310/350
ROR-S	8.360e-6	7.15	0.75 (0.62-0.85)	62/4	456/204
combination	5.602e-8	5.67	0.79 (0.66-0.88)	57/4	392/235
PAM50	3.054e-12	11.25	-	59/7	344/316
scmgene	1.086e-2	1.80	-	31/7	335/188

Table 3.4: Logrank p-values, hazard ratios (HR) and concordance index (c-index) for the different risk scores. Additionally, the number of patients with high/low risk score with and without an event is given. On the top the results for the overall survival (OS) endpoint and on the bottom for the recurrence free interval (RFI) are shown. For the concordance index, the lower and upper bound of the 95%-confidence interval is given in brackets. For all risk scores the low and high risk patients differ significantly in their survival, but overall, EPclin performed best.

(c-index).

Table 3.4 shows these measures for all risk scores. The corresponding Kaplan-Meier plots are available in the Supplement and the interactive report. All risk scores yield significant p-values, hazard ratios well above 1 and a c-index above 0.5. Values above 0.7 are often considered to indicate good prognostic ability for the c-index. For the endpoint *overall survival* (OS), only EPclin yields a c-index above 0.7 whereas the recurrence score, EndoPredict and ROR-S have scores slightly below 0.7. For the *recurrence-free interval* (RFI) however, all risk scores yield c-index scores above 0.7. Interestingly, PAM50 yields a very high hazard ratio and low p-value for the RFI end point. For most endpoints, EPclin performs best: it yields both the lowest p-value and the highest c-index. For the overall survival endpoint, of the 292 patients in the low risk group of EPclin, only 17 had an event, while 74 of the 394 patients from the high risk group had an event after five years. For GGI on the other hand, 28 of 363 low risk patients and 68 of 363 high risk patients experienced an event. The combined risk score, derived from the multivariate Cox regression performs even slightly better than EPclin, with a lower p-value, higher hazard ratio and comparable c-index. However, the effect is moderate, given the increased number of measurements needed.

When only the 370 patients with intermediate risk according to histopathological features (ER+/HER2- patients with grade 2) are considered, ROR-S and GGI perform slightly better than the other risk scores (see Supplement). In this sub-cohort the p-values are generally higher for all risk scores as these patients cannot be classified into low and high risk as easily as the other patients.

For the two subtype classifiers PAM50 and scmgene, the values for the luminal A (low-risk) subtypes are shown. While for PAM50 the luminal A patients have significantly better prognosis, for scmgene the logrank p-value is only 0.001 and also its hazard ratio of 1.48 is by far the lowest of all classifiers.

3.3.3 Concordance of Classifications

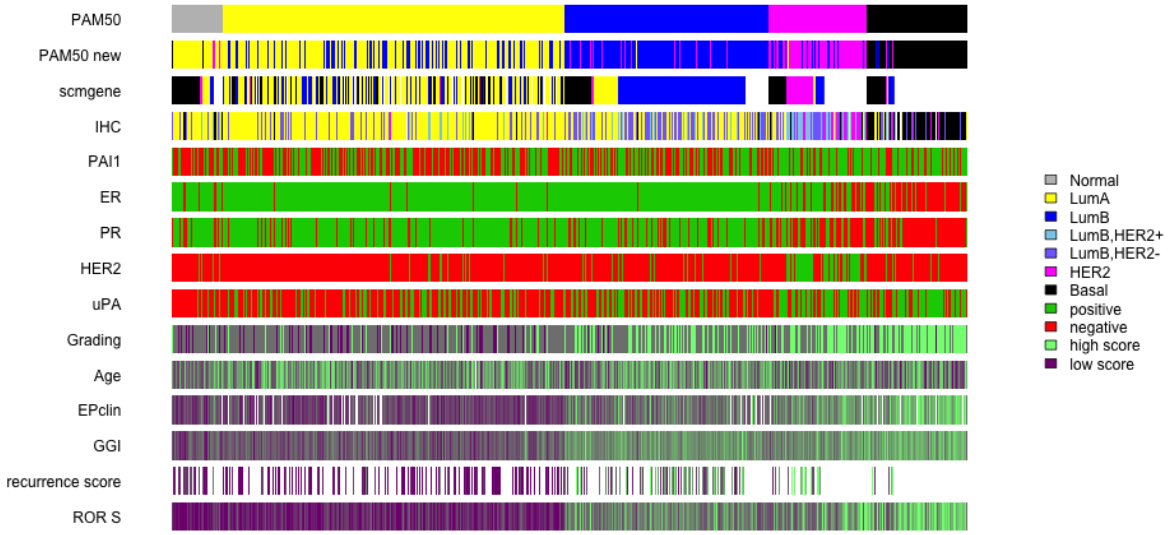


Figure 3.2: Overview of classification results and clinical variables for all patients. The first four rows correspond to subtype classifications, the next 7 rows are clinical characteristics, and the remaining rows are risk scores. A continuous scale between green and purple is used for numeric values such as the risk scores or age and grading and different colors for the categorical attributes. The different subtype classifications are mapped to each other by using prior knowledge (e.g. slightly different names for the luminal A subtype by PAM50, scmgene or the histopathological classification) or by maximizing the overlap to the histopathological classification (for the newly trained PAM50).

Figure 3.2 shows the predictions of all classifiers, as well as some clinical characteristics for all patients. Each row corresponds to one classifier/characteristic and each column corresponds to one patient. The patients are ordered in the same way in all rows (according to PAM50), so that the predictions/characteristics can be compared for each patient. Both variants of PAM50 (using the published model (PAM50) or training a new model (PAM50 new)) yielded similar results. The main difference is that the newly trained model only returns 4 subtypes, so that the normal-like subtype is missing. The predicted subtypes are in many cases the same as the histopathological subtype, only for HER2 overexpressing and luminal B patients, the two classifications differ. The predictions of scmgene that only uses three genes to predict the subtype differ in many cases from the prediction of PAM50. Especially the normal-like patients are predicted to be basal according to scmgene, while the newly trained PAM50 classifies them as luminal A. These patients are assigned a low risk score by all other methods and they are ER positive and HER2 negative according to the immunohistological measurements. Also, only 2 of the 19 patients had an event within five years, so these are likely false predictions of scmgene.

All the risk scores predict predominantly low risk scores for the patients that had luminal A or normal-like subtypes, and high risk scores for the basal and HER2 subtypes according to PAM50. Their predictions differ most for the luminal B patients. Here, GGI and EPclin predict high scores for most patients, while EndoPredict and the recurrence score yield mostly low scores. The recurrence score did not return a risk score for many patients, as it uses 21 genes, and cannot return a result if a measurement for any of these genes is missing.

	recurr. score	**EP**	**EPclin**	**GGI**	**ROR-S**
recurr. score	0.991	0.602	0.563	**0.718**	0.577
EP		0.997	0.626	0.536	0.506
EPclin			0.997	0.524	**0.473**
GGI				0.997	0.614
ROR-S					0.997

	PAM50	**PAM50 new**	**scmgene**	**histopath.**
PAM50	1.000	**0.837**	0.484	0.478
PAM50 new		1.000	0.486	**0.578**
scmgene			1.000	0.419
histopath.				1.000

Table 3.5: Cramer's V for risk scores (top) and classifiers (bottom). Most risk scores and classifiers correspond well to each other. The highlighted values are discussed in the text.

Table 3.5 shows the Cramer's V statistic for the risk scores and subtype classifiers. All risk scores correspond quite well to each other, with Cramer's V values above 0.5, which indicates strong association. Only the comparison of ROR-S and EPclin yielded a Cramer's V slightly below 0.5. The recurrence score and GGI are most similar according to the Cramer's V statistic, yielding a value of 0.718. The concordance of the subtype classifiers was inferior to the risk scores. Only the published and newly trained PAM50 classifiers corresponded well to each other, while scmgene only yielded Cramer's V statistics of 0.484 and 0.486. We also compared the subtype classifier's predictions to the clinical histopathological subtypes. The newly trained PAM50 had the highest correspondence with these clinical subtypes, yielding a Cramer's V value of 0.58, while scmgene again yielded the least correspondence with a Cramer's V of 0.419.

3.3.4 Robustness to Noise

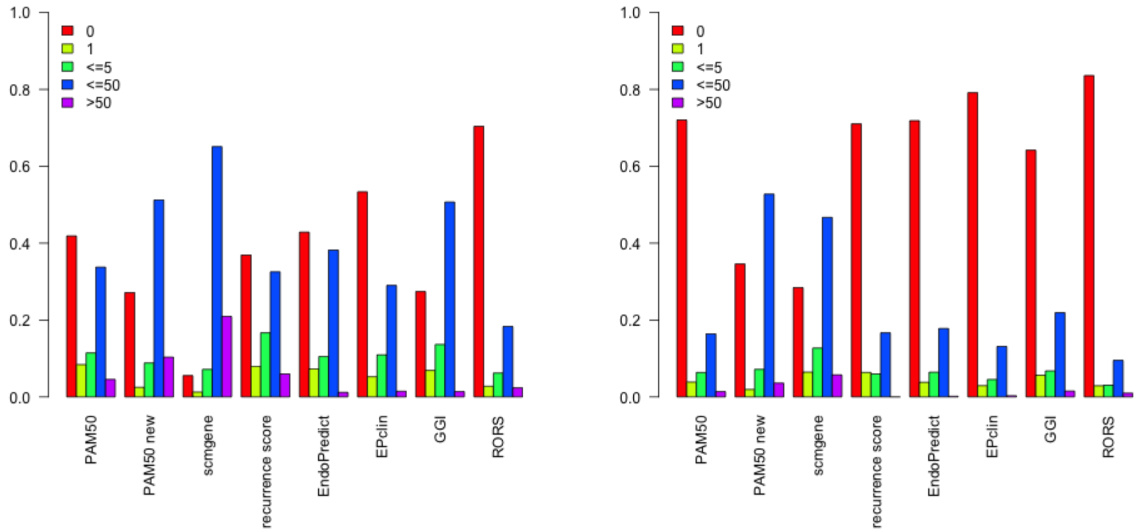
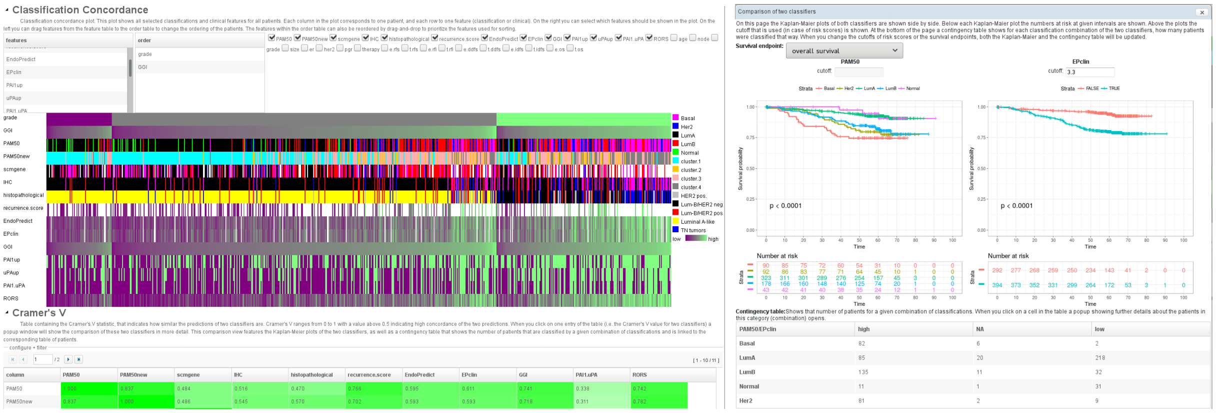


Figure 3.3: Number of patients with a given number of misclassifications for each classifier when noise sampled from $\mathcal{N}(0, 0.7)$ (left) and $\mathcal{N}(0, 0.3)$ (right) is added to the measurements.

To analyze the robustness of the classifiers to experimental noise, we simulated 100 datasets where we added a small noise term to each measurement, and compared the resulting prediction to the predictions without noise. The left plot in Figure 3.3 shows for each classifier how many patients were misclassified how often in the 100 runs, using a normal distribution with mean 0 and sd 0.7 ($\mathcal{N}(0, 0.7)$) as noise distribution. The ROR-S score performed best, with 506 patients without any misclassification. Interestingly, PAM50 with a newly trained model seems to overfit and yields for many patients different predictions when noise is added. Only 219 patients were never or only once misclassified. Similarly, scmgene is very sensitive to noise and yields different predictions for nearly all patients: only 44 patients were never or only once misclassified. The robustness to noise does not seem to depend on the number of genes used by the classifier, as e.g. the recurrence score that uses 21 genes, performs worse than EndoPredict that uses only 7 genes. It might rather depend on the way the gene expression measurements are used or which genes are selected by the classifier.

We repeated this simulation using a smaller noise term sampled from a $\mathcal{N}(0, 0.3)$. The newly trained PAM50 and scmgene still yielded many misclassifications for most patients.



Classification Concordance

Classification concordance plot. This plot shows all selected classifications and clinical features for all patients. Each column in the plot corresponds to one patient, and each row to one feature (classification or clinical). On the right you can select which features should be shown in the plot. On the left you can drag features from the feature table to the order table to change the ordering of the patients. The features within the order table can also be reordered by drag-and-drop to prioritize the features used for sorting.

features	order
EndoPredict	grade
EPclin	GGI
PAI1up	
uPAup	
PAI1.uPA	

☑ PAM50 ☑ PAM50new ☑ scmgene ☑ IHC ☑ histopathological ☑ recurrence.score ☑ EndoPredict ☑ EPclin ☑ GGI ☑ PAI1up ☑ uPAup ☑ PAI1.uPA ☑ RORS ☐ age ☐ node ☐ grade ☐ size ☐ er ☐ her2 ☐ pgr ☐ therapy ☐ e.rfs ☐ t.rfs ☐ e.rfi ☐ t.rfi ☐ e.ddfs ☐ t.ddfs ☐ e.idfs ☐ t.idfs ☐ e.os ☐ t.os

Cramer's V

Table containing the Cramer's V statistic, that indicates how similar the predictions of two classifiers are. Cramer's V ranges from 0 to 1 with a value above 0.5 indicating high concordance of the two predictions. When you click on one entry of the table (i.e. the Cramer's V value for two classifiers) a popup window will show the comparison of these two classifiers in more detail. This comparison view features the Kaplan-Meier plots of the two classifiers, as well as a contingency table that shows the number of patients that are classified by a given combination of classifications and is linked to the corresponding table of patients.

configure + filter

1 / 2 [1 - 10 / 11]

column	PAM50	PAM50new	scmgene	IHC	histopathological	recurrence.score	EndoPredict	EPclin	GGI	PAI1.uPA	RORS
PAM50	1.000	0.837	0.484	0.516	0.470	0.766	0.595	0.611	0.741	0.338	0.742
PAM50new	0.837	1.000	0.486	0.545	0.570	0.702	0.593	0.593	0.718	0.311	0.782

Comparison of two classifiers

On this page the Kaplan-Maier plots of both classifiers are shown side by side. Below each Kaplan-Maier plot the numbers at risk at given intervals are shown. Above the plots the cutoff that is used (in case of risk scores) is shown. At the bottom of the page a contingency table shows for each classification combination of the two classifiers, how many patients were classified that way. When you change the cutoffs of risk scores or the survival endpoints, both the Kaplan-Maier and the contingency table will be updated.

Survival endpoint: overall survival

PAM50 cutoff:

EPclin cutoff: 3.3

Strata	0	10	20	30	40	50	60	70	80	90	100
Basal	90	85	75	72	60	54	31	10	0	0	0
Her2	92	86	83	77	71	64	45	10	1	0	0
LumA	323	311	301	289	276	254	157	45	3	0	0
LumB	178	166	160	148	140	125	74	20	1	0	0
Normal	43	42	41	40	38	35	24	12	1	1	0

Strata	0	10	20	30	40	50	60	70	80	90	100
FALSE	292	277	268	259	250	234	143	41	2	0	0
TRUE	394	373	352	331	299	264	172	53	3	1	0

Contingency table: Shows that number of patients for a given combination of classifications. When you click on a cell in the table a popup showing further details about the patients in this category (combination) opens.

PAM50/EPclin	high	NA	low
Basal	82	6	2
LumA	85	20	218
LumB	135	11	32
Normal	11	1	31
Her2	81	2	9

Figure 3.4: Screenshots of the iReport. On the left the concordance plot sorted by grade and GGI (top two rows) is shown. The sorting can be modified interactively so that the plot can be used to compare different features. On the right the comparison of EPclin and PAM50 with both Kaplan-Meier plots and the contingency table is shown. The cutoff used to separate high and low risk patients of EPclin can be adapted and the contingency table is linked to a table showing all available features for the patients in a specific cell.

The other risk scores, however, became comparable to ROR-S, except that they still yielded more patients that were misclassified in more than half of the noisy datasets.

Moreover, we calculated for each patient and classifier, how much each individual gene would have to differ to change the prediction. The probability of observing noise at least that high can be calculated if a given noise distribution (e.g. $\mathcal{N}(0, 0.7)$) is assumed. These probabilities range from 0 (for measurements that would have to be changed a lot to alter the prediction) to ∼0.6 for our cohort and are available in the interactive report. This way, measurements that are very susceptible to noise can be identified and if possible replicate measurements can reduce the impact of noise for these measurements.

3.3.5 Interactive Report

Figure 3.4 shows two screenshots of the iReport. The screenshot on the left is part of the overall view that shows a summary of the main results discussed in the paper. It shows an interactive version of the concordance plot of Figure 3.2. The user can select which features are included in the plot and by which classifiers the patients should be ordered. This allows to compare several features at once. In Figure 3.4 the patients are ordered first by the tumor grade and then after the GGI risk score that was developed to determine the grade by gene expression. The corresponding two rows are shown at the top of the plot. As can be seen there is some concordance between the two features, with patients with low grade (purple block on the left in grade row) have predominantly low GGI scores, and patients with high grade (green block on the left) have higher GGI scores. However, the majority of patients have intermediate grade and these patients show a distribution of both, high and low, GGI scores.

The screenshot on the right of Figure 3.4 shows the comparison view for PAM50 and EPclin. It contains the two Kaplan-Meier plots side by side and a contingency table below. The cutoff of EPclin that is used to divide the patients into low and high risk can be modified and the corresponding Kaplan-Meier plot will be updated accordingly. The contingency table shows how many patients are classified by the different combinations of subgroups of the two classifiers. The numbers in this table are linked to the corresponding list of patients, so that by clicking on them a table containing all available features of the patients is shown. This way subsets of patients can be analyzed in more detail. E.g. by clicking on the corresponding entry in the contingency table, all information for the 82 patients that were classified as luminal A by PAM50 and high risk by EPclin is shown. This allows the user to look at the survival status of these patients and see that only 11 of these 82 patients are still alive after five years which justifies the high risk prediction of EPclin.

3.4 Discussion

The Fluidigm IFC platform allows to measure the expression of many genes for many patients at rather low cost and with little effort. In this paper we showed that it can be used to measure the gene expression of the genes required for several breast cancer signatures in a large cohort, which enabled us to systematically compare and evaluate these classifiers. For a smaller set of five genes we measured the expression also on a different qPCR platform and the results showed a good agreement between the different platforms after normalization.

The comparison of the classifiers showed that they all performed well on our independent cohort. This shows that the classifiers do not overfit for the cohort on which they were trained but that they are applicable also using a different methodology (Fluidigm) and this new cohort. They provide good estimates of the risk of recurrence of the individual patients. Also their predictions were highly concordant, which also explains why a combined risk score that integrates several classifiers yielded only a slightly better performance.

Moreover, we analyzed the robustness of these classifiers with respect to noise by simulating noise measurements by adding a random noise term. The results showed that especially the classifiers that are newly trained on each cohort, like scmgene or a newly trained model using the PAM50 algorithm, are very sensitive to noise. This also indicates that the cohort that is used to train a new classifier must be of very good quality as noisy measurements can greatly impair the quality of the classifier. Furthermore, also between the classifiers with a fixed model there were large differences in their robustness to noise, as e.g. GGI yielded the same prediction for all 100 noisy measurements only in half as many patients as ROR-S. Furthermore, this kind of noise analysis can also be used to attribute each measurement with a probability that noise changes the prediction for a given patient. This can be used to identify measurements for additional replicates to reduce the impact of noise.

It has to be noted that our unselected cohort was comprised of patients with relative good clinical prognostic factors. Those HER2 positive or receptor negative cases which received neoadjuvant chemotherapy were not included since fresh frozen material has not been available. The classifiers perform differently on cohorts with higher proportions of these patients. In this work we demonstrated feasibility to analyze a large number of genes by PCR and use the publicly available research versions of the classifiers on that same cohort. Second, because we used the research versions of the classifiers and not the commercial versions the results may differ slightly. Third, we were unable to include information on therapy which undoubtedly had an effect on outcome.

All the results of this paper are also available as interactive report (iReport) on the accompanying website in order to make all results reproducible and transparent. This website allows to analyze the results and especially the differences between the classifiers in much more detail as is possible in a paper. The online tool allows selection of cases, strata, classifiers, endpoints and visualization of results. Cross-sectional comparison of clinical and histopathological data and classifiers assigned to each patient can be seen. Longitudinal data is shown as Kaplan-Meier curves as by defined groups. Thus on the one hand, the iReport provides an easy to use interface to results that cannot be shown in a paper due to page limitations, as e.g. the Kaplan-Meier plots for all classifiers for all survival endpoints. On the other hand, it also includes much more detail for individual results by linking the raw data to the summarized result, as is e.g. done by showing the patient lists with all available data for the contingency table of the classifications of two classifiers. This is also important for individualized medicine, where a comprehensive visualization of the individual measurements that are considered for the therapy decision is crucial. We believe that this detailed data can help to generate new hypotheses, e.g. about the patients that are discordantly classified and can thus help the further development of new classifiers.

Chapter 4

RelExplain - Integrating Data and Networks to Explain Biological Processes

Motivation

The following three chapters describe methods to create models that help to understand biological systems. These models can have various forms and explain the biological system on different *context-levels*. In this chapter we focus on subnetworks that show the relations between genes that are changed in the measurement. This is a model that is very easy to understand when the subnetwork is not too large.

A typical analysis workflow of high-throughput experiments is to determine the differential genes/proteins and to characterize them by doing an enrichment analysis. These methods use functional annotation to determine which processes contain more changing genes than expected by chance. However, they do not provide information about how the genes within the biological process interact and whether additional genes may be involved. This is the result of significant area search methods, but as they do not incorporate functional annotation the subnetworks that they return are unspecific and do not necessarily correspond to a biological process of interest.

RelExplain bridges the gap between these two approaches. It combines experimental data, networks and process information to return an explanation: the optimal subnetwork that connects the differential genes in a given biological process. To calculate this explanation it takes the consistency between the type of the edge and the changes of its adjacent nodes as well as the functional annotation of the nodes into account. The resulting explanations are compact networks of the relevant part of the process and additional nodes that might be important for the process and can easily be interpreted.

Our evaluation showed that RelExplain is better suited to retrieve manually curated subnetworks from unspecific networks than other algorithms. The interactive RelExplain tool allows to compute and inspect sub-optimal and alternative optimal explanations.

Publication

The content of this chapter is published in Bioinformatics [9]. Here, it is reformatted and parts of the supplement are integrated.

Author Contributions

Evi Berchtold analyzed the data, implemented and evaluated the method and wrote the manuscript. Gergely Csaba and Evi Berchtold designed the method. Ralf Zimmer supervised the project and edited the manuscript.

Availability:

A webserver to calculate the RelExplain explanations is available at

`https://services.bio.ifi.lmu.de/relexplain`

4.1 Introduction

High-throughput experiments such as microarrays or RNAseq are usually done differentially to compare the gene expression between two or more experimental conditions, and one is interested in the differences between these conditions. Standard statistical preprocessing and analysis determines a set of differentially expressed genes DG. To better understand the differences, affected biological processes are identified using the DG. The ultimate goal is to understand how these involved processes determine the different phenotypes and the measured (differential) data. However, genome-wide high-throughput experiments often yield long lists of differential genes. The in detail analysis of many differential genes is time-consuming at best and the overall interpretation of DG is difficult.

Therefore, gene set enrichment methods are used to determine which processes are associated with a predefined set of so called *terminal nodes*, typically the differential genes, more often than expected by chance. There are various methods available for this task (reviewed in [60, 61]). Overrepresentation analysis (ORA) approaches test whether the genes of DG are associated with a biological process more often than expected as quantified via a hypergeometric test. The disadvantage of these types of methods is that genes have to be classified beforehand as differential or not differential and that the cutoff used is somewhat artificial but can have a large impact on the results. This problem is addressed in approaches such as Gene Set Enrichment Analysis (GSEA) [92], which ranks the genes by their fold change or p-value and uses a Kolmogorov-Smirnov statistic to assess the significance of the gene set.

Enrichment methods yield ranked lists of pathways or processes that are overrepresented for the given experimental data. While this can yield interesting and unexpected insights which processes are involved in the changes between the experimental conditions, it usually is only the first step of the analysis. Often, one is interested in a certain aspect of the experiment or it is known beforehand from the design of the experiment, from previous experiments, or from previous prior knowledge, which processes are important, but one is more interested in the mechanistic details how the genes interact within the process and how consistent the interactions are with the measured data and evidence.

If one is interested in the details of how the genes interact, an underlying network and network search methods can be used (reviewed in [72]). These methods find subnetworks that contain many differential genes. The subnetworks are often subsequently tested for enriched processes so that it is possible to find subnetworks enriched for a process or a combination of processes which can give insights in how these processes are connected. However, there is no method that takes prior knowledge of an involved process explicitly into account. So, no focused analysis of a specific process is possible if one is interested in a certain aspect of the experiment, but one has to hope that a subnetwork enriched for the process of interest is among the returned top scored subnetworks. Moreover, the resulting subnetworks are often quite large and difficult to interpret.

SteinerNet [53] finds the optimal prize-collecting Steiner tree, that is it determines the tree with minimal edge distance that connects most terminal nodes. For SteinerNet, edge distances are derived from the reliability of the edges. In the prize-collecting variant of the Steiner tree problem not all terminal genes have to be included, but the prize of adding an edge is balanced against the cost of omitting a terminal gene.

Another much-used network search method is jActiveModules [55]. In this method, subnetworks are scored by an aggregated z-score that indicates how much the genes in the subnetwork deviate from the overall distribution of expression scores in the experiment. A simulated annealing approach is then used to find high-scoring subnetworks.

HotNet2 [68] is a recent method that is based on network propagation. The experimental measurements are used as heat scores that are than propagated along the edges to the neighboring nodes. "Hot" subnetworks are then returned as the interesting subnetworks.

More and more gene set enrichment methods that take network information into account to score and rank the BPs (reviewed in [73]) have been proposed. GGEA [39] is one of these methods that is based on a notion of consistency in the network, which quantifies the compatibility of the measured data with the edge types. But also GGEA first of all delivers a network score, which is used to rank the processes in question.

Here, we propose RelExplain, a method that is designed to analyze a particular biological process bp in the context of a given network to unravel the relevant relationships of the involved genes in the process. A typical workflow would be to identify interesting processes by enrichment methods and then analyze them in more detail with RelExplain. RelExplain returns a connected subnetwork that contains most differential genes within the process and, if necessary, further genes to connect them. To select these genes various aspects such as the corresponding experimental data and their annotated processes are taken into account. The interactions in the subnetworks can be used as a starting point for new hypotheses that may be validated in further targeted experiments.

For a semantically meaningful and, thus, interpretable explanation it is crucial to provide a mapping between the kind of measured data and the type of interactions and relations in the network. This mapping will then enable to define reasonable measures of plausibility, consistency, and interestingness for edges, genes, and whole subnetworks, e.g. RelExplain solutions and/or biological processes. Altogether these measures should not only allow for a better quantification whether certain biological processes are affected according to the measured data, but also to provide detailed insights into which edges and regulations of target genes are compatible with a pathway hypothesis or at least which edges are interesting in one way or the other (consistent or inconsistent) with a network hypothesis given the actual measured data.

Furthermore, as there are often multiple similarly optimal or suboptimal subnetworks, we provide an interactive tool to inspect alternative paths in the subnetwork. While minimal solutions provide compact representations of how the genes interact, a biological pathway needs not to be minimal, but will contain redundant paths. Using the interactive RelExplain tool one can find high-scoring alternative paths and decide whether they should be included in the subnetwork or not.

4.2 Methods

4.2.1 Network

RelExplain allows to use (directed and undirected) networks compiled from various sources, such as textmining edges, protein-protein interactions (PPI), gene regulation networks or post-translational modifications. Each edge between two nodes can consist of several edge instances, if it is derived from several sources. An edge can, e.g. be a PPI and a phosphorylation and would be represented by two edge instances, one from the PPI database and one from the post-translational modification source. Each edge instance is annotated with its type (e.g. gene regulation), its reliability (e.g. manually curated database) and its source (e.g. YEASTRACT).

Whereas RelExplain is designed to use heterogeneoues data and network types, here we use only regulatory transcription factor : target gene (TF:TG) networks from YEASTRACT [93] and RELEX [35] textmining edges.

As standard of truth networks we employ hand-curated networks for the diauxic shift in yeast as assembled from Geistlinger et al [38].

RelExplain uses the networks to compute scores for a biological process *bp* based on the nodes in the *bp* and the edges between them in order to produce a compact interpretable representation of them via visualized networks. Internally, RelExplain keeps track of types and any additional annotations to edges and nodes. These additional information can be queried and visualized via the interactive RelExplain tool.

4.2.2 Assigning distances to edges

Most network-based methods use not only the edges, but also data associated to them such as the length of an edge (distance) and/or its reliability (p-values or other measures of statistical significance). E.g. finding the best Steiner tree asks for computing a tree with the shortest length/distance connecting a set of predefined set of (terminal) nodes. Given an edge from any input network with a specific type, the associated distance should reflect how good the edge semantically fits the measured data in the experiment. Moreover, edges can be inside the *bp* (both its nodes are in the analyzed *bp*), connecting the *bp* to the outside (one node in *bp*, the other not), or outside (both nodes are not contained in the *bp*). RelExplain tries to connect the terminal nodes from the *bp* (e.g. the differential genes contained in *bp*) via edges within the *bp* with an as small as possible overall distance. If necessary, outside nodes are also used and even outside edges, both incurring a respective penalty.

Scoring of nodes

We first score the nodes in the given input network such that genes that should be included in the RelExplain solution, because they are differential in the data or belong to the analyzed biological process *bp* (or a similar one), receive a high score. For expression data as used here, the absolute fold change (fc) is used as node score. Alternatively, if p-values for the differential expression of the genes are available they can be used instead of absolute fold changes to score the nodes and edges. In contrast to absolute fold changes, a low p-value (and not a high absolute fold change) indicates a differential gene. To calculate the edge score, the two p-values are combined by Fisher's method. The combined node score for an edge is calculated subtracting the process penalties of both nodes from $-\log_{10}$ of the combined p-value. To favor genes that belong to *bp* or a closely related process, we subtract a process penalty from the score of the gene node. The process penalty for a process P is defined via the Jaccard distance $d(P, bp)$ of P with *bp*. For any gene node G annotated to some processes P_i, we subtract the minimum distance of any P_i to the biological process *bp* in question ($\text{penalty}(G) = \min_i d(P_i, bp)$). If subtracting the penalty would produce a negative score, a score of 0 is assigned to the node.

Gene nodes that belong to *bp* receive a process penalty of 0, while genes that are annotated to processes that share no genes with the analyzed process receive the maximal penalty of 1.

Scoring of edges

Next, we assign a score to each edge in the network. For edges that consist of multiple edge instances (that is they are derived from multiple sources), each edge instance is scored separately and the best, i.e. highest, score is used for the edge. First, the combined node score n is defined as the mean score of the two adjacent nodes.

For expression data and gene regulatory edges with annotated sign (the edge is activating or inhibiting) a consistency score c similar to GGEA is used. An activating edge is consistent if the source and target are changing in the same direction, whereas an inhibiting edge is consistent if they are changing in opposite directions. The impact of this score should depend on how much the target gene of the regulation is changing (quantified by the fold change) as this indicates the impact of the regulation. Therefore, we define the

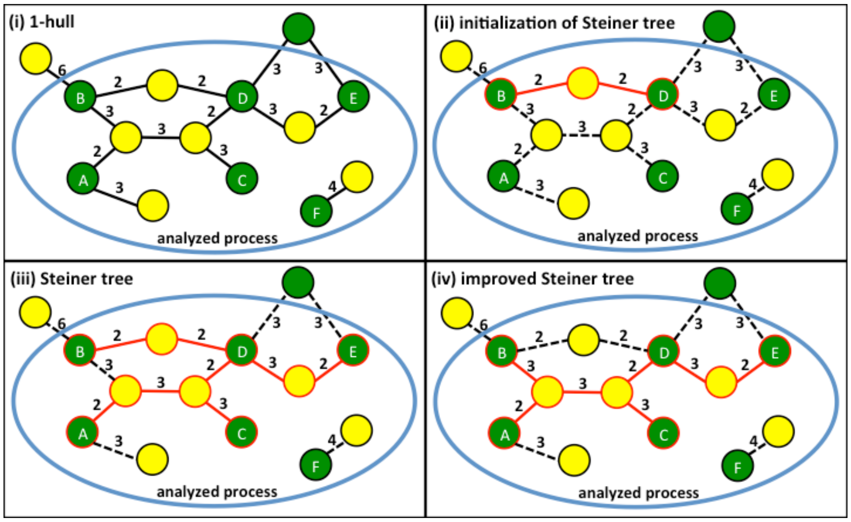

(i) 1-hull
(ii) initialization of Steiner tree
(iii) Steiner tree
(iv) improved Steiner tree
A
B
C
D
E
F
6
2
3
4
analyzed process

Figure 4.1: Schematic visualization of the RelExplain algorithm. The terminal nodes (differential genes contained in the analyzed biological process) are shown in green (nodes A-F), the current Steiner tree is highlighted in red. The network is first restricted to the d-hull (here d=1, see subfigure (i)). The Steiner tree is initialized with the shortest path between two terminal nodes (here path between B and D, see subfigure (ii)). Iteratively, the remaining terminal nodes are connected to the Steiner tree by their shortest path to any node in the current Steiner tree (subfigure (iii)). For each non-terminal node, it is checked whether it is necessary to connect the terminal nodes in the graph induced by the Steiner tree. Thereby, the node between B and D could be removed from the Steiner tree.

consistency score to be the node score of the target if the edge is consistent and -1 times the node score if the edge is inconsistent.

As the network used by RelExplain consists of edges from various sources, the reliability of the edges varies. Edges that are derived from textmining or from high-throughput experiments are more error-prone than edges from manually curated databases or low-throughput experiments. Depending on the evidence for the edge, a reliability score r can be assigned according to Table 1 in the Supplement.

The final score of the edge s_{edge} is a weighted combination of the node score n, the consistency score c and the reliability score r:

$$s_{edge} = (\max(0, w_n * n + w_c * c)) * (1 + w_r * r) \tag{4.1}$$

The weights of the subscores (w_n, w_c and w_r) can be adjusted by the user to reflect the relative importance of these factors, as default $w_n = w_c = 0.45$ and $w_r = 0.1$ are used.

To calculate distances (edge lengths) from these scores, the maximal score of all edges is determined and the difference of the edge score to this maximal score is used as distance.

4.2.3 RelExplain Steiner tree approximation

The goal of RelExplain is to explain the interactions for the measured data within a given biological process *bp*. A set of nodes in *bp* is designated as terminal nodes. In this paper we define the terminal nodes as the differential genes in the analyzed process *bp*. A RelExplain explanation is a Steiner tree that connects all terminal nodes (if possible within *bp*). A Steiner tree is a tree with minimal edge distance that connects all terminal nodes, but can also contain non-terminal nodes.

As the Steiner tree problem is NP-complete [58] we use an approximation to find the Steiner tree. The complexity of this approximation scales with the size of the network. As we do not want to include long paths containing many insignificant nodes, in a preprocessing step we restrict the network to the *d*-hull around the terminal nodes. The parameter *d* indicates how many non-terminal nodes are allowed on a path between two terminal nodes.

The approximation starts with the shortest path between two terminal nodes, which can be computed using the Dijkstra algorithm (Fig. 4.1 (i)). In each subsequent step, the shortest path connecting a not yet connected terminal node to any node in the growing Steiner tree is added (Fig. 4.1 (ii)). When all terminal nodes are added to the Steiner tree (Fig. 4.1 (iii)), we check for each non-terminal node whether there is an alternative path in the tree such that the induced graph is still connected without this node (Fig. 4.1 (iv)). Note that this simple heuristic procedure always constructs a tree as cycles cannot occur. The final improvement step prunes superfluous nodes if a connected tree can be produced with fewer non-terminal nodes.

On the other hand, alternative paths of similar length can, as an option, also be included into the final RelExplain solution. The iterative addition of shortest paths to connect terminal nodes is geared at paths inside *bp* but, depending on the edge length, can also include nodes and edges outside the *bp*.

RelExplain is a heuristic and very fast: RelExplain adds the terminals one at a time via a fast procedure, a Dijkstra search starting from one terminal node to another node already in the Steiner tree. Thus, the overall worst case complexity of RelExplain is $O(|TN| * (|E| + |V| log|V|))$, where TN are the terminal nodes, and V and E are the nodes and edges of the used network. In practice, due to the locality and possible "small world"-features of the network, RelExplain is much faster.

4.2.4 Finding alternative paths

Often small variations of a subnetwork yield almost optimal scores, but most methods only report the best-scoring subnetwork. Similarly, RelExplain heuristically aims at computing the Steiner tree with the smallest distance. But as mentioned above, the tree can be extended by alternative paths of the same (or similar) distance. Of course, these paths only increase the overall length of the solution, but may add relevant explanations within the biological process *bp* closely connected to the rest of the solution. Moreover, as the used scores, networks, and experimental data are not perfect, suboptimal paths may also be important for the biological interpretation of the solution network. Therefore, RelExplain allows to add these slightly suboptimal variations to the solution in order to include redundant regulations into the solution network that are missing in the optimal Steiner tree.

However, if all alternative variations are included in the subnetwork, it can become rather large and, thus, hinder its interpretation. Typically, RelExplain allows only for alternative paths which score very close to the optimal one (paths having a distance at most ϵ times longer than the optimal distance). RelExplain ϵ, however, shows that RelExplain is quite robust with respect to alternative and ϵ optimal paths, which implies that redundant regulations typically remain quite compact (instead of yielding very large solutions covering larger parts of the whole network).

In the interactive mode of RelExplain, the user can select any two terminal nodes and find all suboptimal paths between those nodes with any user-defined deviation from the optimum. These redundant suboptimal paths are found by a breadth-first search keeping track of all paths from the start node until the error threshold or the end node is reached. Again, this procedure is very fast. It is also fast to add all ϵ-suboptimal paths instead of only the optimal path during the approximation of the Steiner tree.

4.2.5 Evaluation using manually curated subnetworks

The evaluation of network-based methods is challenging as there are no comprehensive gold standard networks or methods to simulate data realistically. The best way is to use curated networks for well-studied processes even though they do not necessarily have to be complete.

Geistlinger et al. [38] manually curated context-dependent subnetworks for the diauxic shift in yeast. Overall, the diauxic shift network is partitioned into eight subnetworks, such as gluconeogenesis, glyoxylate cycle or TCA cycle. These subnetworks are supposed to contain all edges that are relevant for the different subprocesses during the diauxic shift and, thus, can be used as a gold standard for RelExplain. The typical input for network methods, however, are biological networks that contain interactions/regulations for several conditions. Thus, these generic networks also contain nodes and edges that need not be active in the analyzed condition. Network methods should be able to extract relevant subnetworks for the specific context (such as the diauxic shift subnetworks) from larger networks with many more irrelevant edges.

In our evaluation, we use the manually curated diauxic shift gluconeogenesis subnetwork as gold standard to investigate whether RelExplain and other methods can reproduce these networks given experimental diauxic shift data [24] and a network that contains the edges from the gold standard network and additional unspecific decoy edges. As terminal nodes we choose the set of differential genes, i.e. genes with an absolute $\log_2$ fold change larger than 1.

To use a realistic setup for the decoy edges, we choose two real networks, the textmining network RELEX (9.129 additional edges between 2.849 nodes) and the gene regulatory network YEASTRACT (35.393 additional edges, 6.191 nodes), and randomized their edges. As especially gene regulatory networks such as YEASTRACT have a special degree distribution with few transcription factor nodes (hubs) with many outgoing edges and many target genes with few incoming edges, we used a rewiring procedure, which keeps both the hubs and the degrees of the nodes invariant.

Even for the diauxic shift network and diauxic shift data only a small fraction of the nodes in the subnetworks are actually differential in the experimental data. As network methods aim at identifying subnetworks with many differential nodes they are unable to identify the complete gold standard gluconeogenesis network, which contains many unchanged nodes. Thus, to make their task easier, we define the gold standard as follows: We start with the terminal nodes ($|fc| > 1$) that are contained in the Geistlinger gluconeogenesis subnetwork. The edges of the gold standard are all edges of the Geistlinger network connecting these nodes. This network is then extended by additional nodes and edges in order to minimally connect the individual components of the gold standard network. If there are multiple alternative non-terminal nodes that could be used to connect terminal genes all alternative gold standards are considered.

We applied RelExplain, SteinerNet, jActiveModules and HotNet2 using standard parameters and, furthermore, a variant of RelExplain that includes all alternative and suboptimal paths while building the Steiner tree with error margin $\epsilon = 1\%$. jActiveModules needs p-values, but as no replicates are available for the DeRisi data, we calculated p-values from the z-scores. HotNet2 returned many very small subnetworks. Thus, for the evaluation we also considered the combination (union) of all subnetworks that contained at least one terminal node, even though these combined solutions are not necessarily connected.

SteinerNet, jActiveModules and HotNet2 have no information concerning the process that should be analyzed and use the whole network and associated experimental data as input. Therefore, we also applied adapted (+)-versions SteinerNet+/jActiveModules+/HotNet2+ that use only the genes within the analyzed process as input. SteinerNet uses as input a network and a set of terminal nodes. For SteinerNet(+) the terminal nodes are restricted to the differential nodes in *bp*. jActiveModules and HotNet2 use experimental data and a network as input. For their respective (+)-variants we restrict the experimental data to *bp*, so that only genes in *bp* are provided with a fold change/p-value and all genes that are not contained in *bp* are considered as unmeasured.

All applied methods return a solution network consisting of a subset of the nodes of the input network and all edges between these nodes. The performance of the respective methods is assessed based on the included nodes. For each resulting solution network the f-measure with respect to the gold standard is calculated and if multiple networks are computed, the solution with the highest f-measure is used.

4.2.6 Application to TCGA data

To demonstrate the versatility of RelExplain, we also applied it to a set of 106 breast cancer patient data from TCGA [18] for which both tumor and normal tissue samples were measured via RNAseq. For each patient data local fold changes [27] between tumor and normal samples were calculated and enriched GO categories were identified using hypergeometric and Kolmogorov-Smirnov tests. The GO category "ERK1 and ERK2 cascade" was enriched for 90% of all patients. Therefore, we selected this category as an example biological process *bp* to be analyzed in more detail by RelExplain.

In this example, we use the median of the fold changes over all patients for the node

method	network	f-measure	|overlap gold|	|solution|	|nodes in process|
RelExplain	Diauxic	1.000	19	19	19
RelExplain ϵ	Diauxic	1.000	19	19	19
SteinerNet	Diauxic	1.000	19	19	19
SteinerNet+	Diauxic	1.000	19	19	19
jActiveModules	Diauxic	0.364	12	47	17
jActiveModules+	Diauxic	0.400	5	6	6
HotNet2	Diauxic	0.438	7	13	9
HotNet2+	Diauxic	0.789	15	28	28
RelExplain	Diauxic,RELEX	1.000	19	19	19
RelExplain ϵ	Diauxic,RELEX	0.974	19	20	20
SteinerNet	Diauxic,RELEX	0.056	16	553	21
SteinerNet+	Diauxic,RELEX	1.000	19	19	19
jActiveModules	Diauxic,RELEX	0.161	14	155	20
jActiveModules+	Diauxic,RELEX	0.320	4	6	4
HotNet2	Diauxic,RELEX	0.357	5	9	5
HotNet2+	Diauxic,RELEX	0.655	19	39	39
RelExplain	Diauxic,RELEX,YEASTRACT	0.865	16	18	17
RelExplain ϵ	Diauxic,RELEX,YEASTRACT	0.865	16	18	17
SteinerNet	Diauxic,RELEX,YEASTRACT	0.048	17	683	21
SteinerNet+	Diauxic,RELEX,YEASTRACT	0.889	16	17	16
jActiveModules	Diauxic,RELEX,YEASTRACT	0.028	17	1197	32
jActiveModules+	Diauxic,RELEX,YEASTRACT	0.182	2	3	3
HotNet2	Diauxic,RELEX,YEASTRACT	0.286	5	16	5
HotNet2+	Diauxic,RELEX,YEASTRACT	0.600	12	21	21

Table 4.1: Results of the diauxic shift evaluation for the gluconeogenesis subnetwork. For each combination of method and network the f-measure and the number of nodes that are shared with the gold standard are shown. Furthermore, the size of the solution and how many nodes are annotated to the analyzed process is given.

score. Genes for which the complete confidence interval of the local fold change lies above 0.2 or below -0.2 in at least 70% of all patients were defined as terminal nodes. As network we used RELEX [35] text mining edges from PUBMED abstracts that contain the term "breast cancer".

We also applied SteinerNet to this data and network. As probability for the edges we used $1 - p$ where p is the p-value of the hypergeometric test of the node co-occurrences in the breast cancer context compared to the background. From the resulting subnetwork, the nodes associated with the "ERK1 and ERK2 cascade" were identified, highlighted and compared to the solution identified by RelExplain.

4.3 Results

4.3.1 Diauxic shift data

The network we use is the hand-curated comprehensive diauxic shift network by Geistlinger et al [38]. The unique feature of this network is that it focused on a specific biological process (the diauxic shift in yeast) and, thus, as one of very available few examples, can serve our evaluation purposes here. Moreover, standard and well-studied experimental data [24] is available as well as high-quality representations of biological processes in question (here gluconeogenesis as curated by Geistlinger et al.). This subnetwork of the diauxic shift network is then restricted to the smallest subnetwork that contains all differential genes to provide an interpretable concise gold standard.

This setup is then employed to assess whether different methods including RelExplain can provide reasonable explanations of the experimental evidence in the investigated context given networks with a mixture of unspecific decoy and the curated diauxic shift edges. Overall, we evaluated 8 different methods on 3 different background networks, altogether 24 approaches. The results are summarized in Table 4.1. As comparison we used three standard network analysis tools: jActiveModules [55], HotNet2 [68] and SteinerNet [53]. They cover quite heterogeneous approaches to the problem. Note that none of these methods is directly able to solve our problem. Therefore, we used the (+)-versions of these tools to improve their results towards reproducing the intended gold standard. In addition, we also list the performance of the ϵ variant of RelExplain adding the ϵ suboptimal and alternative paths to the minimal RelExplain solution.

Table 4.1 lists for each method and network the f-measure, the overlap with the gold standard, the size of the solution, and the number of the nodes in the solution contained in the gluconeogenesis subnetwork.

The first block shows the results of the 8 methods applied to the edges within the diauxic shift network only. RelExplain and SteinerNet perfectly reproduce the gold standard for this artificial setup. Surprisingly, jActiveModules includes several nodes outside the gold standard into the solutions and excludes others. Using the (+)-version of jActiveModules a quite small subnetwork with only 6 nodes is returned. HotNet2 also yields an incomplete subnetwork that misses several important factors of the process. The solution of HotNet2+ is larger, but still not all nodes in the gold standard are covered (15 out of 19). Overall, the overlap with the gold standard as quantified by the f-measure drops from 1.0 to about 0.4.

In the second block, the methods are given the edges of the gluconeogenesis subnetwork and, in addition, the randomized edges of the RELEX text mining network (more than 9.000 edges). RelExplain and SteinerNet+ are again able to reconstruct the gold standard in this case. The HotNet2+ solution contains the gold standard but adds another 20 nodes outside the process to its solution (f-measure = 0.65). Both jActiveModules and SteinerNet return huge networks with 155 and 553 nodes, respectively. Of course, these networks (f-measure of 0.161 and 0.056) would be hard to interpret even though they contain most (but not all!) of the gold standard nodes (14 and 16 out of 19). These methods are not designed to identify the subnetwork that best explains a given process and do not employ process annotations. Thus, they return subnetworks that contain many differential genes that are not contained in the gluconeogenesis process. Again, jActiveModules+ and HotNet2 return only few nodes and, thus, only a very small part of the gold standard solution. The f-measure drops to about 0.06.

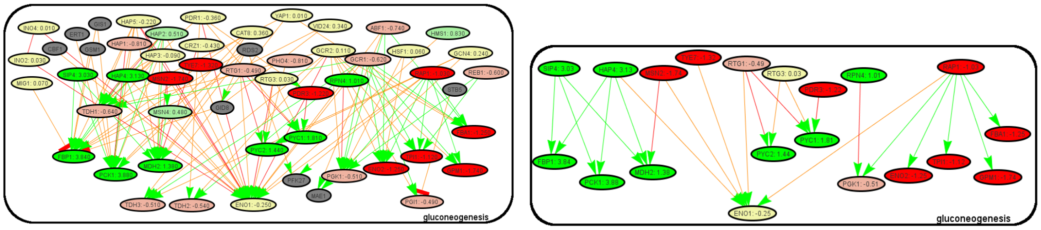
If also the randomized edges of YEASTRACT (>35.000 edges) are added, the results are qualitatively similar. Again, SteinerNet+ and RelExplain perform best, but are no longer able to perfectly reconstruct the gold standard (3 nodes are missing). HotNet2 is the only method for which the (+)-variant yields larger solutions than the normal variant. Apparently, HotNet2+ ignores all nodes without measurement, so that the solutions are restricted to *bp*. Given the complete data, HotNet2 returns many very small subnetworks, that remain unconnected if merged. jActiveModules+ yields a very small solution of little use. The original unrestricted versions SteinerNet and jActiveModules result in very large solutions with 683 and 1.197 nodes with tiny f-measures, respectively.

As an illustration of the results, Fig. 4.2 shows the obtained networks for all methods. The networks are shown in similar layout and the terminal nodes are colored bright red and green depending on their fold change. Yellow nodes do not change significantly. Nodes outside the process or with only moderate fold changes are colored similarly but with transparent colors. Also edges are colored: green edges are consistent (edge type corresponds to observed experimental data), red edges are inconsistent, whereas orange edges indicate cases which cannot be evaluated either due to the type of the edge or the available data. As the used networks only contain gene regulatory edges, an edge is consistent if its sign fits to the changes of the adjacent nodes (as defined for the consistency score) or if the sign of the edge is unknown and both adjacent genes are changing (in any direction). Edges with unchanged genes (yellow) cannot be evaluated and are thus colored orange.

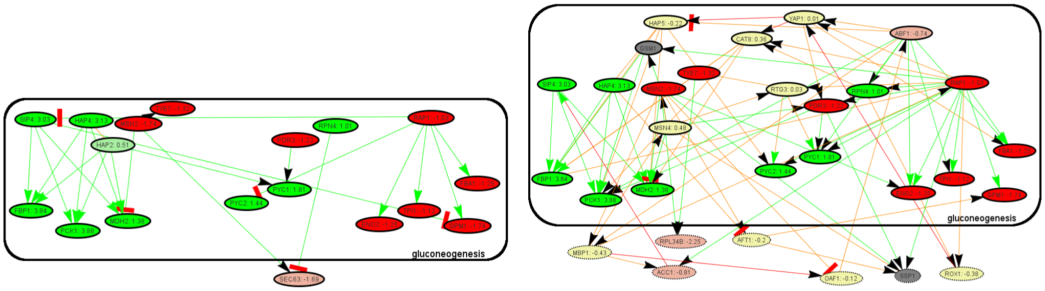
Fig. 4.2 (a) shows the overall gluconeogenesis process as taken from Geistlinger et al (2013). Fig. 4.2 (b) contains the gold standard network extracted from (a) and the experimental data via the definition above (see Methods). Fig. 4.2 (c+d) display the RelExplain and RelExplain ϵ solutions as computed for the most realistic setup with the 45.000 randomized edges (RELEX and YEASTRACT). As can be seen, both solutions exhibit most of the gold standard and its most important factors and regulations. The remaining networks in Fig. 4.2 (e-j) contain the SteinerNet, jActiveModules and HotNet as well as their adapted (+)-version solutions. As can be seen, SteinerNet+ (f) computes a reasonable solution with good overlap with the gold standard, but other solutions are highly unfocused (e+g), small (h), or fragmented (i), which would prohibit a useful explanation of the experimental evidence in the context of the gluconeogenesis.

Both SteinerNet+ and RelExplain include transcription factors that are not contained in the gold standard. As SteinerNet+ gets only the experimental data of the genes within the gluconeogenesis subnetwork as input, it can select these genes only because of their connectivity in the network while RelExplain also takes their process annotation and experimental data into account. As a result, RelExplain selected two TFs with an absolute fold change above 0.5 while SteinerNet+ selects an unchanged TF. RelExplain favors TFs with consistent regulations, which are likely biologically meaningful. Thus, all edges in the RelExplain solution are consistent with the measured data (colored green) whereas SteinerNet+ also contains edges with unknown status (orange edges, Fig. 4.2 (e+f)).

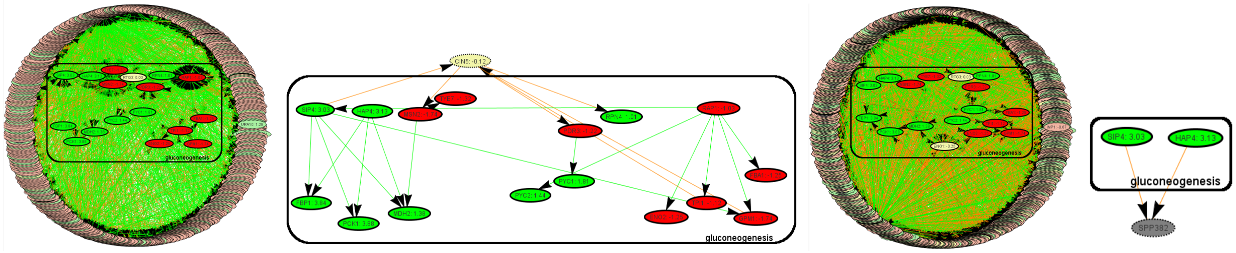
Optimal subnetworks are often not realistic as they are minimal while biological networks exploit redundant paths. To take this into account, RelExplain offers the possibility to search for alternative paths with similar score. This mode yields larger (i.e. more sensitive) solutions that may have a larger overlap with the (by construction minimal) gold standard, but due to the added genes the f-measure is smaller compared to the normal RelExplain run. In any case, the ϵ variants are quite robust as they increase the solutions only moderately. RelExplain solutions are, thus, useful starting points for interactive exploration of explanations including alternative, redundant paths.



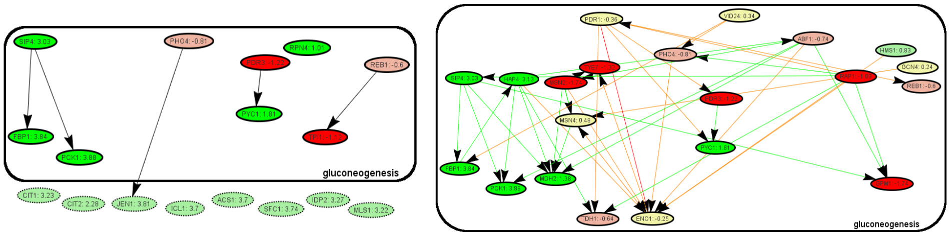
(a) whole process (b) gold standard



(c) RelExplain (d) RelExplain ϵ



(e) SteinerNet (f) SteinerNet+ (g) jActiveModules (h) jActiveModules+

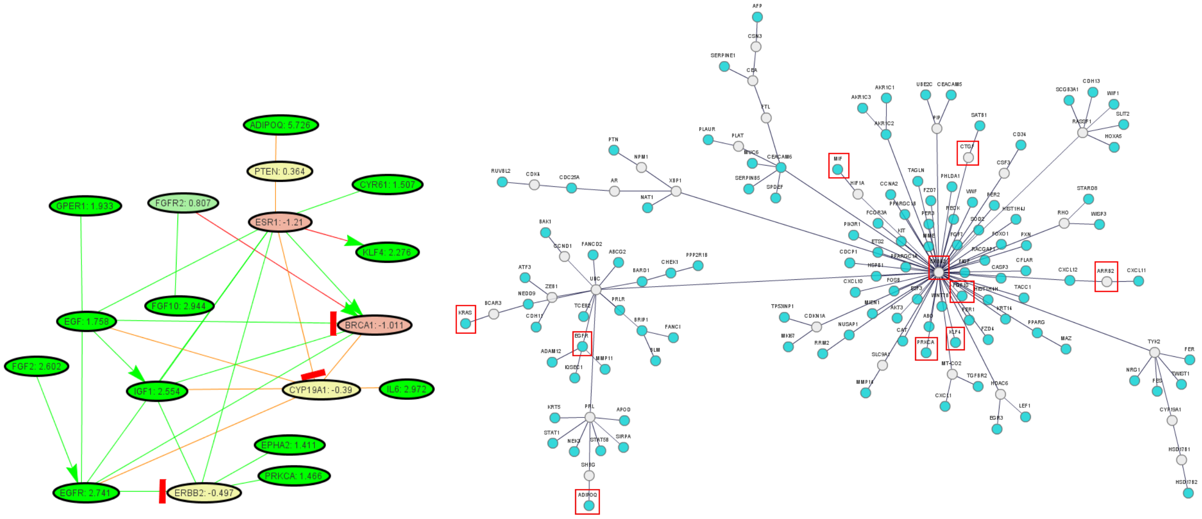


(i) HotNet2 (j) HotNet2+

Figure 4.2: Results of the different methods for the Diauxic,RELEX,YEASTRACT network. Genes that are not contained in the gluconeogenesis subnetwork are dashed, differential genes within the gluconeogenesis subnetwork are colored bright green/red depending on whether they are up/downregulated. Arrows with a green/red tip are known to be activating/inhibiting.

4.3.2 Breast cancer data

RelExplain is a fast and pragmatic method for network analysis and network search towards constructing interpretable explanations. As assessed on standard microarray expression



(a) RelExplain (b) SteinerNet

Figure 4.3: Results for the breast cancer data and the ERK1 and ERK2 cascade for RelExplain (a) and SteinerNet (b). For SteinerNet the genes that are contained in $\mathrm{GO}_{\mathrm{ERK}}$ are highlighted in red. Genes are colored red/green/yellow depending on whether they were up/downregulated/unchanged. Brightly colored nodes are contained in the ERK1 and ERK2 cascade. Arrows with a green/red rip are known to be activating/inhibiting and the color indicates whether they are consistent (green), inconsistent (red) or cannot be evaluated (orange).

measurements and a quite simple complete regulatory system in yeast with a fairly complete and accurate context-dependent network (the "diauxic-shift network" of Geistlinger et al). RelExplain exhibits clear deficiencies of other methods and appears to provide plausible network models as explanations of measured experimental data sets. Here, we report on the application of RelExplain to a larger set of human sequencing data sets from TCGA and investigate its explanatory power on the well-studied ERK signaling pathway.

We analyze NGS data sets from the TCGA compendium [18] for which cancer and normal conditions have been measured for 106 breast cancer patients. As gold standard networks and biological processes are not available we use GO categories instead and compute explanations in the GO class "ERK1 and ERK2 cascade" ($\mathrm{GO}_{\mathrm{ERK}}$) as this GO category was enriched in nearly all patients. We applied both RelExplain and SteinerNet, which performed reasonably for the diauxic shift data in yeast. The resulting subnetworks are shown in Fig. 4.3. The solution of RelExplain consists almost exclusively of terminal nodes that are part of the selected category $\mathrm{GO}_{\mathrm{ERK}}$ (bright green nodes). The genes that were unchanged or not part of the ERK cascade process are known breast cancer relevant genes such as BRCA1, ERBB2, or ESR1. Among the upregulated genes are many growth factors such as EGF, FGF2, IGF1 and FDF10, which are also all associated with the ERK cascade. Most of them are connected to the growth factor receptor EGFR, which can activate the ERK phosphorylation cascade, ultimately leading to proliferation. EGF is known to be released from the plasma membrane after GPER1 activation by estrogen and can then itself activate EGFR [32]. As the regulations downstream of EGFR along the ERK signaling cascade are on the phosphorylation level the corresponding genes need not change in the expression data. The expression data does not yield useful information on these regulations and, thus, they are not included in the RelExplain subnetwork explaining the RNAseq data.

SteinerNet has been applied to compute the Steiner tree for all terminal nodes and all nodes that are associated with the category GO_{ERK} are highlighted (red boxes). The resulting tree is quite large and cannot be easily interpreted. Some of these highlighted genes are only connected to the other genes via long paths (e.g. KRAS is connected by a path of length 5 to EGFR). Altogether, the SteinerNet solution contains only 6 terminal nodes (blue nodes) that are highlighted in Fig. 4.3. The known regulation of EGFR by EGF and GPER1 is missing as well as most upregulated growth factors. As compared with the RelExplain solution SteinerNet both (i) includes more nodes into its solution and (ii) misses many factors and regulations important for the biological process in question. Obviously, based on the SteinerNet solution an in-detail interpretation of the breast cancer data in the context of the ERK signaling process is hampered. Although many downstream phosphorylation events cannot (in principle) be observed in the data and are, thus, missing in the solution, the RelExplain network yields a much simpler explanation of the data at hand and yields a compact representation of the experimental evidence for the ERK signaling process. An overview of the results is given in Supp. Table 1.

4.4 Discussion

Set enrichment methods identify the relevant sets of genes from a collection of candidate sets via appropriate statistical tests. If this collection is derived from a set of pathways or processes they can also identify those pathways which appear to be deregulated in a given experiment based on a statistically significant number of differential genes. Typically, these methods do not evaluate or explain how exactly the involved genes interact and regulate each other.

On the other hand, pure network approaches ignore the functional annotations, so that the resulting subnetworks are not limited to a particular biological process. This leads to large and unfocused solutions which severely impair the interpretation and usefulness of the results. Obvious workarounds would be, in a postprocessing step, to restrict the solution to the nodes (and thus edges) that are annotated to the process. The restricted solutions can, however, be suboptimal as the process information was not used during the construction of the solution such that important factors for a meaningful explanation of the process can be missing. E.g., two terminal nodes in the process could be connected by a node that does not belong to the process even though there exists another node in the process that also connects those nodes and would make more sense within the used process context.

We propose RelExplain, a method that takes both, the network and the functional annotations (processes) explicitly into account to explain a given experiment in a process context. RelExplain identifies subnetworks that are specific for a given process so that a more focused interpretation is possible. The functional annotation is utilized in the scoring of the edges so that nodes within the process context or within similar processes are included preferentially.

RelExplain solutions strive to produce minimal Steiner trees connecting the terminal nodes to enable a focused interpretation. However, these solutions biologically need not be minimal, but often several alternative and/or redundant paths exist and can also be simultaneously active in biological processes. Depending on the context and the data, these alternative paths can score as good or only slightly worse than the optimal paths and, therefore, they are presumably also of interest for a biological interpretation. Most network based analysis methods aim at producing optimal, i.e. minimal solutions, so that these alternatives necessarily have to be omitted (e.g. in optimal Steiner trees). To enable both a focused and a more holistic interpretation RelExplain can compute minimal solutions, as well as extended solutions containing alternative paths and suboptimal paths (which are only suboptimal by a small margin ϵ).

Moreover, RelExplain provides an interactive mode in which all alternative and suboptimal paths as well as neighborhoods of the solution can be explored together with the experimental evidence as given by the respective experimental data. We expect that using this mode one can explore the solution space and get a more comprehensive view of the genes involved and their exact role in the process as indicated or supported by the experimental data.

The approach and implementation of RelExplain allows to easily incorporate alternative scoring schemes. Thus, it can not only be used for expression data as shown in this paper, but also for other types of genome-wide data e.g. ChIP, DNase footprinting, time series data, proteomics, ...) and even several heterogeneous datasets at the same time. An example application to genome-wide data observed for a cohort of individual breast cancer patients is briefly described above. RelExplain only requires that a score can be calculated from the data that measures whether a given edge of a certain type is interesting (supported by the data in the experimental context) and, thus, should be included in the resulting subnetwork.

When several measured data sets and several edge types are available, a match between data set and edge type is required in order to define meaningful scores for the implication of the respective edge on the actual data. As examples, DNA-array or RNA-seq expression data can be used to define implications of an active transcription factor on its target genes (but tells nothing about phosphorylation edges), or, PPI or textmining co-occurrence edges can be used to score co-expression of the connected nodes.

4.5 Conclusion

It is a major goal to model biological processes and mechanisms on a level that allows the accurate simulation of the process and to make predictions on its perturbation. A less far-fetching goal is to interpret sets of large-scale measurements in the context of such biological processes in order to assess whether the data sheds light on its workings. According to our experiments the current gene set enrichment methods and the network based analysis methods are not sufficient for any of these goals. Even if set enrichment methods would be perfect in identifying the relevant sets of genes, their further use for the analysis of data and/or biological processes is very limited and the user is left alone with these tasks. Literally hundreds of publications stop at this point of printing long lists of "statistically significant" GO categories. Network based enrichment methods use more prior knowledge to improve the ranking of the relevant categories while network search methods use a given network to provide more insights into the internal structure of the data but without using the functional annotation. But there is no way to combine already obtained enrichment results into the network analysis, as the network search methods only use the experimental data and networks as inputs. Our experiments show that current network search methods have severe limitations to really mechanistically interpret the data and a biological process as they lack detail or focus, or both at the same time.

RelExplain is a simple significant area search method, which allows to compactly assemble and represent the evidence of the measured data for the prior knowledge available on a given biological process *bp* in question. RelExplain can work with different kinds of networks and several sets of heterogeneous measurements and integrates them into a concise network model for *bp*. This model via the RelExplain score and via visual inspection allows to directly assess the available evidence in the context of the available prior knowledge. RelExplain is algorithmically simple, very fast and can work with very large networks. It is robust in the sense that the resulting models are compact and focused to the process in question, but at the same time not excluding possible alternative or redundant paths. Moreover, the RelExplain models serve as entry points for interactive in depth analysis of both the underlying networks and the analyzed measured data. This is facilitated via extending the network by alternative and suboptimal paths as well as exploring network neighborhoods all, of course, in the context of the available experimental data.

Chapter 5

Modeling of the Changes during Yeast Heat Shock Response

Motivation

The following chapter describes an integrative approach that, in contrast to RelExplain, also provides quantitative predictions. It is used to model the protein abundance changes over time in response to a mild and severe heat shock in yeast.

The central dogma of biology states that the DNA is transcribed into mRNA, that is in turn translated to build proteins which fulfill some function in the cell. In many cases, however, it is not that simple and further mechanisms impact one or more steps in the dogma. New high-throughput methods such as ribosome profiling allow to measure the outcome of the individual steps. Integrating these new techniques with other high-throughput datasets can be used to model the changes in a system to identify additional regulatory mechanisms that might deviate from the central dogma or influence the efficiency of the individual steps.

Here, we analyze data on three different stages of the central dogma: gene expression, translation by ribosome profiling and protein levels. The gene expression and translation of many genes is changed for both the mild and severe heat shock, but the corresponding protein levels remain similar to the unstressed cell.

To analyze whether this is nevertheless consistent with the central dogma or if some additional regulatory mechanism is needed to explain this inconsistency, we modeled the changes downstream of the gene expression both qualitatively and quantitatively. The most parsimonious fit was achieved when an increased degradation for translationally upregulated and decreased degradation for translationally downregulated proteins was assumed. This would indicate that the altered protein stabilities are compensated for by the changed gene expression and subsequent translation to achieve protein homeostasis.

Publication

The contents of this chapter have not been published yet. A manuscript focusing on the biological implications of the described data and results is in preparation.

Author Contributions

Christopher Stratil performed the microarray measurements and prepared the samples for the unfractionated proteomic measurements. Moritz Mühlhofer prepared the samples for the fractionated proteomics measurements, and together with Christopher Stratil performed the ribosome profiling measurements. Nina Bach performed the proteomics measurements. Stephan Sieber supervised the proteomics measurements. Martin Haslbeck and Johannes Buchner supervised all other measurements.
Gergely Csaba preprocessed the ribosome profiling and proteomics data. Evi Berchtold analyzed all data and performed the integrated analysis and modeling. Ralf Zimmer supervised the data analysis and modeling.

5.1 Introduction

Adapting to a suddenly changed environment or stress is a crucial ability of all organisms. Metabolic processes need to be adapted to the new conditions to function optimally and detrimental effects have to be mitigated. Unicellular organisms need an especially fast stress response as they are in direct contact with the changing environment instead of being contained in the relatively stable environment of a tissue or organ. The response to heat of Saccharomyces cerevisiae is one of the best studied stress response systems.

A number of physiological changes of the yeast cells occur when the temperature is increased above the optimal growth temperature of 25-30°C: The cell cycle of yeast cells is arrested in the G1 phase, the cell wall and membrane dynamics change and proteins aggregate as they are misfolded [97].

Already in 1998 and 2000, the first high-throughput microarray measurements were done by Eisen et al. [25] and Gasch et al. [36] to analyze the expression changes upon various types of stress. Most of the analyzed types of stress showed massive changes in the gene expression affecting hundreds of genes. Using a hierarchical clustering, they could show that a large set of genes show a similar pattern of activation in different types of stress. These genes can be divided into 300 stress-activated and 600 stress-repressed genes and are together called the environmental stress response (ESR). The repressed genes are involved in various growth related processes, like RNA metabolism and nucleotide biosynthesis, and ribosome protein genes. In contrast, the activated genes were often uncharacterized or involved in carbon metabolism, detoxification of reactive oxygen species, cellular redox reactions, cell wall modification, protein folding and degradation, DNA damage repair, fatty acid metabolism, metabolite transport, vacuolar and mitochondrial functions, autophagy, and intracellular signaling. For heat shock specifically, they observed that many chaperones and genes involved in respiration and alternative carbon source utilization were changed.

In 2011, Lee et al. [66] analyzed the protein changes of yeast upon osmotic stress and found that there is a poor correlation between downregulated proteins and their gene expression level. This could be due to longlived proteins, as the old protein persist even though fewer new proteins are produced. But the effect is also observed for proteins with short half-lives. They used a modeling approach to simulate protein levels using expression levels, absolute protein quantifications and protein half-lives. Using this simulation they could determine that including the observed growth arrest in the model resulted also in nearly constant protein levels for downregulated transcipts. A similar study was done for oxidative and heat stress in fission yeast by Lackner et al. [65].

In 2013, Shalgi et al. [85] found a different level of regulation. They analysed ribosome profiling data of heat shock in mouse and found a global pause in translation elongation. More specifically the ribosomes are stalled after translating ∼65 amino acids which corresponds roughly to the length of the ribosome tunnel. They showed that the chaperone Hsp70 that normally associates with ribosomes and folds newly synthesised proteins, does not associate with the ribosome upon heat which could result in a stalling of the translating ribosome. Similarly, Lui et al. [69] showed a Hsp70 dependent stalling of ribosomes at the same position upon proteotoxic stress in yeast.

Here, we want to model the effects downstream of the transcriptional changes. We analyze gene expression, ribosome profiling and proteomics data together to unravel the downstream effects of the changes in gene expression. For this we model the system, first qualitatively by comparing up- and downregulated genes in the different datasets, and then more detailed also quantitatively by integrating data measuring protein half-lives, absolute protein amounts and growth and fitting a model similar to Lee et al.

5.2 Data

To analyze the changes upon heat shock we measured the system on different levels: gene expression measurements capture the changes in transcription, ribosome profiling measures the changes in ribosome-bound mRNA that is likely to be actively translated, and proteome measurements estimate the amount of proteins, not only in total but for one time point also separately for the soluble and insoluble fraction.

To capture the dynamics of the response for all measurements time series were done. Furthermore, two different strengths of the stress were applied: a mild heat shock at 37°C, and a more severe heat shock at 42°C. For the gene expression measurements the most comprehensive set of measurements, containing several very early time points as well as later time points (1, 3, 5, 7, 10, 15, 40, 80 and 160 min), is available. This showed that the peak of the gene expression changes is between 10 and 15 min, and for a mild heat shock at 37°C the mRNA levels return back to normal thereafter. In contrast, at 42°C, the changes in mRNA levels persist. For the other types of measurements, we focused on two time points: at the peak of the expression changes after 10 min, and when the expression

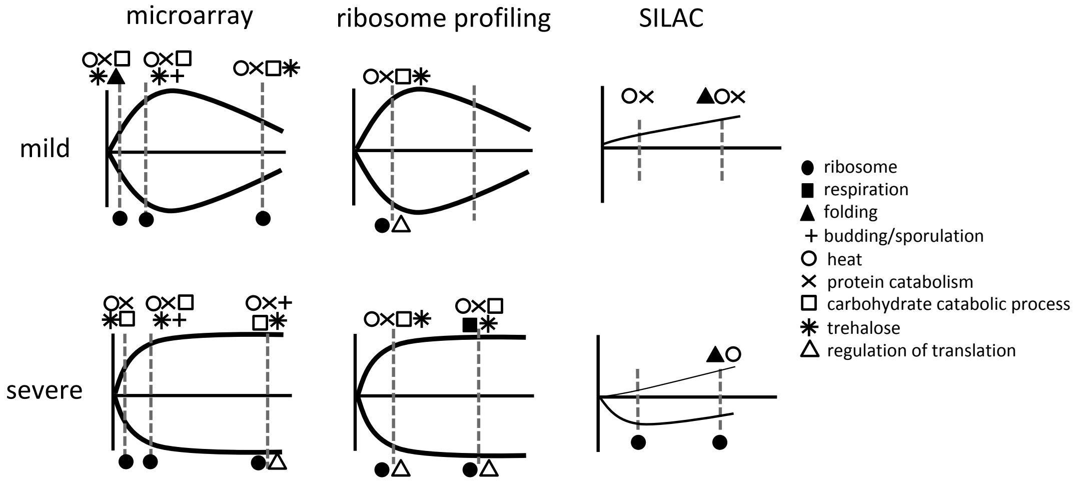


Figure 5.1: Overview of the results for the different datasets. For each dataset the changes over time are depicted schematically. The width of the line indicates how many genes/proteins are changing and the dashed vertical lines indicate the measured time points. Furthermore, enriched processes at different time points are shown by the symbols.

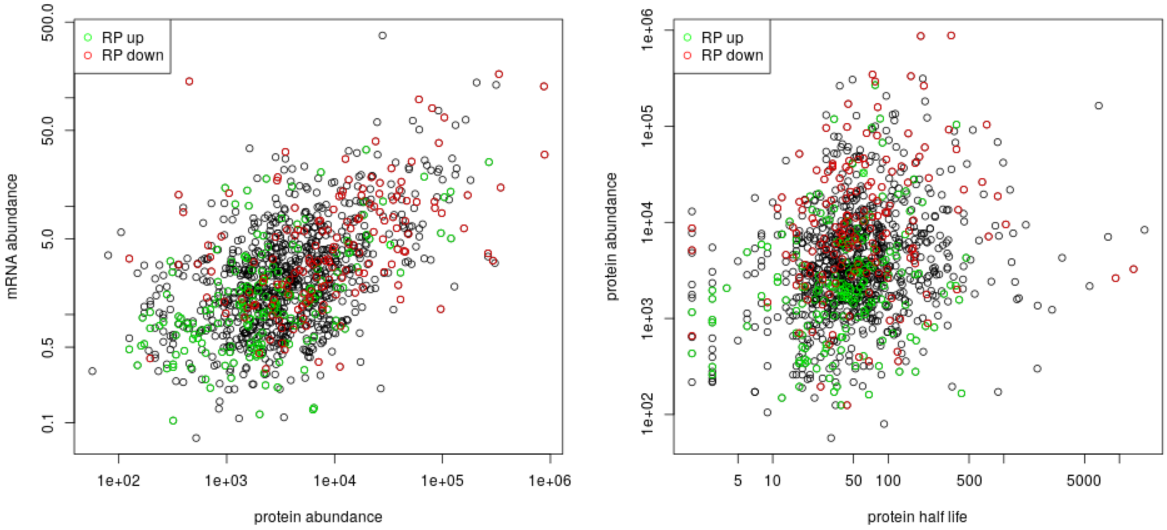
changes already decrease for the mild heat shock after 30 min.

Figure 5.1 shows a schematic overview of the measurements and the progression of the changes over time as well as enriched processes for the individual measurements. The changes in the levels of ribosome-bound mRNA measured by ribosome profiling are very similar to the changes observed in the gene expression measurements. This indicates that the up- and downregulated genes are also translated differentially, so that a change in the protein levels would also be expected. However, there are far fewer changes on the protein levels. At 37°C there are some upregulated proteins, involved in response to heat and protein catabolism, but there are far fewer proteins upregulated at 42°C, even though the changes are more pronounced in the ribosome profiling data. Also there are some downregulated proteins after 10 min, that return back to normal after 30 min, even though the corresponding genes show decreased translation in the ribosome profiling data and stay downregulated after 30 min.

The less pronounced changes on the protein level could be due to the higher absolute protein abundances compared to mRNA abundances and/or long protein half-lives and thus low protein turn-over, which both result in less pronounced protein fold changes. Our quantitative modeling approach takes both mRNA/protein abundances and protein half-lives into account. We used protein half-lives measured by Belle et al.[7], absolute initial protein levels from Ghaemmaghami et al.[42], initial mRNA levels from Miura et al.[74]. Figure 5.2 gives a short overview of this data.

The absolute mRNA abundances were measured using a competitive PCR between genomic DNA and cDNA. It provides estimates of the absolute mRNA levels of 4,416 transcripts which range from 0.0175 to 376,4875 copies per cell.

The absolute protein abundances were measured using quantitative western blots of a



(a) (b)

Figure 5.2: Additional data used in the quantitative modeling with the genes that are up- and downregulating in the ribosome profiling (RP) data highlighted. (a) Protein and mRNA abundances in copies per cell. While a cell contains 1-10 copies of most transcripts, the corresponding proteins occur in much higher concentrations of 1000-10,000 proteins per cell. Nevertheless there is a moderate positive correlation. (b) Protein half-life and abundance. There is no strong correlation between protein half-life and protein abundance.

epitope-tagged fusion library and it covers 5,709 proteins. Here the copies per cell range from 41-1,590,000 proteins. Figure 5.2 (a) shows the comparison of the absolute abundances for the 4,287 genes/proteins for which both kinds of data is available. Even though the protein abundance is, in general, orders of magnitude higher there is a positive correlation. The genes that are upregulated in the ribosome profiling data are in general a little less abundant in both the mRNA and protein level than the genes that are downregulated.

Belle et al. measured protein half-lives by measuring the protein abundance over time upon inhibition of protein synthesis by cycloheximide. The protein abundances are measured using quantitative western blots of an epitome-tagged fusion library. This resulted in half-life estimates for 3,176 proteins, which were approximately log-normal with a median half-life of 43 min. Figure 5.2 (b) show the comparison to the absolute protein abundances. There is no clear correlation: there are also highly abundant proteins with rather short half-lives and lowly abundant proteins with very long half-lives. There is also no clear difference between the protein half lives of proteins with up- and downregulated translation.

5.3 Methods

5.3.1 Quantitative Modeling

Lee et al. [66] proposed a model to predict the protein change $P(\rho, t)'$ of protein ρ over time from the mRNA concentration $m(\rho, t)$, protein degradation rate $k_d(\rho)$ estimated from the protein half-lives and dilution due to growth μ. For each protein the translation rate $k_s(\rho)$ is estimated. The change in the protein concentration in their model is calculated by the following equation:

$$P'(\rho, t) = k_s(\rho)m(\rho, t) - (k_d(\rho) + \mu(t))P(\rho, t) \tag{5.1}$$

Similar to Lee et al. we used published datasets of protein half-lives [7], absolute initial protein levels [42], initial mRNA levels [74] and our own measurements of ribosome profiling, protein fold changes and growth rates. Given these measurements, Lee et al. could estimate the protein synthesis parameter $k_s(\rho)$ for each protein to fit optimally to all time points for which protein levels were measured. During osmotic shock there is growth arrest for about the first 45 min of the stress, so that they found that it was best to fit for each protein two $k_s(\rho)$ parameters, one for the measurements during the growth arrest (30 min) and one for the measurements afterwards (60, 90, 120 and 240 min). In contrast, we only measured protein abundances up to 30 min after the heat was applied and did not see a marked difference in the growth rate during this time. Moreover, as we use ribosome profiling instead of expression data, $k_s(\rho)$ corresponds to the protein production rate from ribosome bound mRNA instead of the translation rate given the mRNA abundance. The rate limiting step of translation is initiation [84] so that we conclude that $k_s(\rho)$ is constant over time and should not change upon heat shock. The unstressed cells should be in a steady state with equal protein synthesis and degradation. This allows us to calculate the synthesis rate $k_s(\rho)$ directly from the equilibrium measurements without the need to fit thousands of parameters which could lead to overfitting:

$$\begin{aligned} P'_\rho(t_0) &= 0 \\ k_s(\rho)m(\rho, t_0) - (k_d(\rho) + \mu(t_0))P_(\rho, t_0) &= 0 \\ k_s(\rho) &= ((k_d(\rho) + \mu(t_0))P(\rho, t_0))/m(\rho, t_0) \end{aligned}$$

However, this model does not take into account that many ribosomes could be stalled similarly as in mouse and do not produce protein with the same efficiency as in unstressed conditions. To predict whether the translation of a transcript is stalled we applied the approach described by Shalgi et al [85]. In short we determine the position with the maximal difference between the sum of all reads up to this position normalized by the total number of reads in the two samples. The significance of this position is then estimated using the Kolmogorov-Smirnov statistic. Only ribosomes downstream of this stalling point will contribute to protein production, so we adapted the ribosome profiling fold changes by ignoring all reads mapped to positions upstream of the predicted stalling position of the transcript. Using these modified ribosome profiling fold changes takes the effects of ribosome stalling into account.

To integrate the increased aggregation of protein upon heat shock, the fractionated proteome measurements can be used. This measurement was only done for 30 min, so in order to model the changes after 10 min, a linear inference is used to infer the proteome measurements for this time point. For the fractionated proteome measurements the soluble (supernatant) and insoluble (pellet) fraction as well as both fractions together (total) were measured. This allows us to analyze the effect of aggregating proteins, that are only present in the total and pellet fractions but not the soluble fraction. Thus, we compare the simulation results using the soluble fraction to the simulation using the total fraction to assess the impact of aggregating proteins.

Additionally to modifying the input data to incorporate ribosome stalling or protein aggregation, we can also fit the protein synthesis parameter k_s or a factor for the measured protein half-lives. The rational behind this is that even though we can estimate the protein synthesis and decay (according to half-life) in equilibrium conditions, the biological system is not in equilibrium upon heat shock and both the synthesis and decay rate could change. Especially a change in the decay rate of proteins makes sense, as the increased temperature leads to increased aggregation and degradation so that the half-lives measured under equilibrium conditions do not apply. We can fit these parameters in different ways: we can fit both the synthesis parameter k_s and the degradation parameter k_d for each protein separately which corresponds to each protein responding differently to the heat shock, or we can identify different groups of proteins that are all modeled using the same parameters.

To fit an individual synthesis rate $k_s(\rho)$ for each protein, we use the least-square error estimate of k_s described by Lee et al. For fitting an individual decay rate $k_d(\rho)$ for each protein, we assume that the measured protein half lives still provide valuable information and thus fitted a multiplicative factor for the measured half life. These two fits require the same number of parameters as there are proteins which makes the method vulnerable for overfitting. A more parsimonious assumption would be that there are groups of proteins that are affected similarly by the changing environment. A natural grouping of the proteins would be according to their changes in the ribosome profiling data: proteins with upregulated translation (fc$>$ 1 in at least one time point), proteins with downregulated translation (fc< -1 in at least one time point) and proteins with unchanged translation ($-1 <$fc< 1 in all time points). In case of proteins that are both up- and downregulated they are assigned to the group in which direction it showed the higher fold change. For each of these groups we fitted a multiplicative factor for the measured half lives yielding the minimal deviation if the simulated and measured fold changes.

In total we test four different variants: (a) no fit, calculate synthesis and decay rate from the unstressed (equilibrium) measurements, (b) grouping of the proteins and fitting of one degradation factor for each group, (c) fitting of an individual half-life factor for each protein and (d) fitting of the protein synthesis parameter k_s similarly to Lee et al.

The supernatant measurement from the fractionated proteome measurement as well as another independent proteome measurement can be used as independent test sets to evaluate the validity of the individual fits. For this we fit the parameters of the fit using the proteome time series and compare the simulated protein fold changes using these parameters with the changes from the independent measurements.

All measurements used in the simulations are associated with biological and technical noise. To estimate the effects of the noise and whether observed discrepancies between simulated and measured protein levels are due to noise, we developed the following simplistic approach to incorporate noise. For both the proteome measurements and the ribosome profiling data replicates are available. We repeat the simulations using all replicate combinations and determine the interval that is covered by the resulting simulated protein abundances. If some parameters are fitted we estimate them using the averaged measurements, so that an unstable fit does not result in huge predicted intervals. To evaluate the simulation, it is checked whether this interval overlaps with the interval of replicate proteome measurements. Note that we cannot estimate the noise of the proteome abundance and half-live measurements as no replicates are available, so that we in general underestimate the overall noise. For proteins/genes with missing values in some of the replicates we use a simple value imputation strategy to obtain comparable intervals. We determine the median standard deviation (sd_{med}) of the replicates over all genes/proteins of the measurement and approximate the missing replicate(s) by the mean of the remaining replicates plus a $\mathcal{N}(0, sd_{med})$ error term.

5.4 Results

5.4.1 Qualitative Modeling

The most simple model of the effects downstream of transcription is the central dogma of biology. It says that the DNA is transcribed into mRNA, which is in turn translated into proteins. In order for gene expression changes to have an effect on the cell, they need to be transformed into protein changes, as proteins are the functional entities in a cell. We can thus compare the measurements corresponding to different steps in the central dogma to each other qualitatively, i.e. whether the genes/proteins change in the same direction.

Figure 5.3 shows such a comparison. It shows both the scatterplot between the fold changes of the two measurements as well as how many genes are contained in specific regions of the plot. The fold changes of each measurements are divided into four different categories: upregulated (fc>1), unchanged trending up (0<fc<1), unchanged trending down (-1<fc<0) and downregulated (fc<-1). The comparisons are shown both after 10 min and 30 min.

In the comparison between RNAseq and ribosome profiling data, most genes are up/downregulated in both datasets, or up-/downregulated in one and unchanged but trending in the same direction in the other dataset. After 30 min there are genes upregulated in the ribosome profiling data that are not upregulated in the RNAseq data, which indicates that their translation rate (i.e. translation per mRNA molecule) increases during the heat shock. Similarly, there are genes with (slightly) downregulated expression that are even more downregulated in the ribosome profiling data. Both these observations suggest that

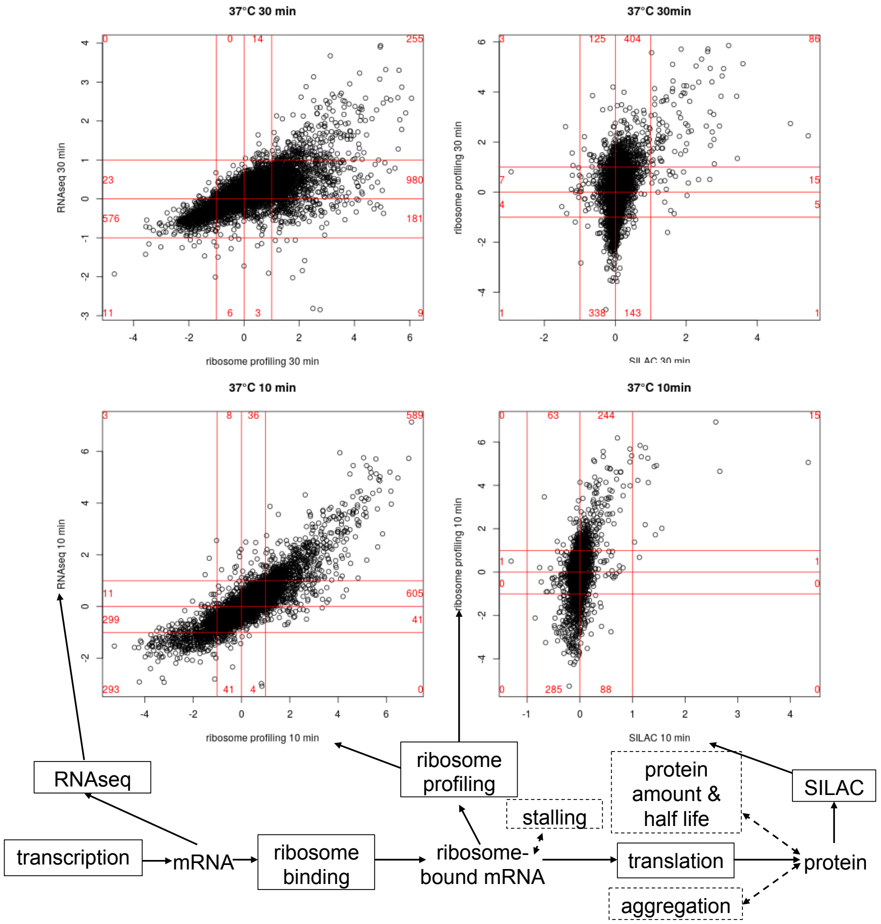
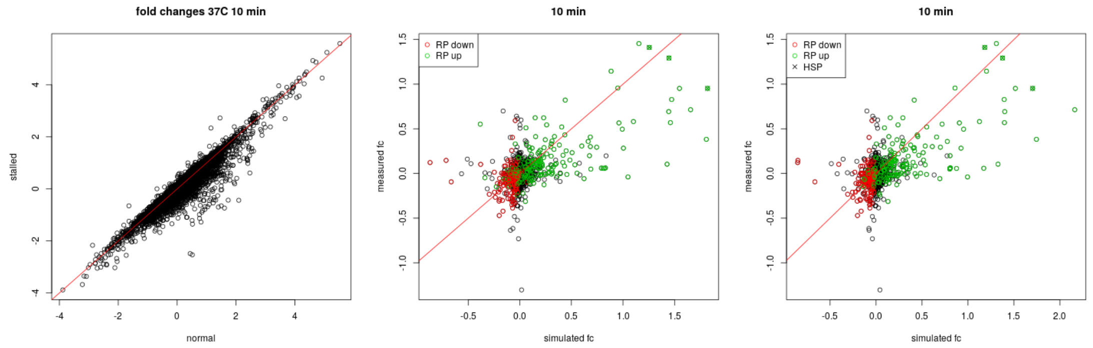


Figure 5.3: Comparison of the changes in the different measurements together with the underlying model. The scatterplots contain in red the number of genes/proteins contained in the corresponding area. While most changes are consistent between mRNA and ribosome-bound mRNA, the protein levels do not fit.

the translation amplifies the regulation of gene expression after 30 min at 37°C. At 42°C, these effects are however not observed, but both ribosome profiling and expression change similarly. Overall, for all analyzed conditions there is a positive correlation between the changes and ribosome profiling, and thus ribosome binding, is consistent with the changes in gene expression.

In contrast, the changes in the proteome measurement fits much worse to the ribosome profiling data. There are nearly no genes/proteins that are up- or downregulated in both datasets. In general the changes in the protein levels are much less pronounced than on the ribosome-bound mRNA. Most of the proteins that are changed in the ribosome profiling data, but unchanged in the SILAC data are trending in the right direction, but there are also much more proteins that are trending in the wrong direction, compared to the



(a) stalled vs normal (b) simulation stalled data (c) simulation normal data

Figure 5.4: Influence of ribosome stalling on the simulation using equilibrium protein synthesis. (a) Comparison of fold changes that take stalling into account to normal fold changes. Many proteins show slightly lower fold changes when stalling is considered. Results of the simulation if stalling is considered (b) or not (c). There are nearly no changes in how well the measured fold changes can be simulated.

comparison of microarray and ribosome profiling data. Due to the different abundances of proteins and mRNAs in the cell, protein changes are expected to be more subtle. Also, as the synthesis of proteins takes time the changes observed in gene expression manifest only later on. Here, however there are still fewer changed proteins after 30 min. It is also possible that the discrepancy between protein levels and gene expression is due to heat shock specific effects, such as increased protein aggregation or stalling of the ribosomes that are bound to the mRNA. The Petri net below the scatterplots in Figure 5.3 shows these different explanations in the dashed boxes. To analyze which effect explains the data best, a more detailed quantitative modeling including all these effects has to be done.

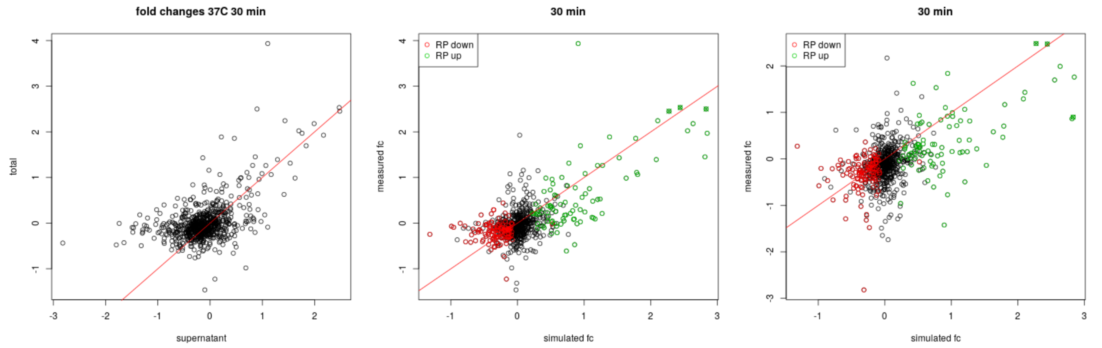
5.4.2 Quantitative Modeling

Influence of Stalling and Aggregation

First, we want to assess how much the model is improved by including the modified data incorporating ribosome stalling and protein aggregation. For this we modeled the protein levels using the protein synthesis rate calculated from equilibrium measurements and the measured half-lives so that no parameters need to be fitted that could mask the differences between the different input datasets.

Figure 5.4 (a) shows how taking into account ribosome stalling changes the resulting fold changes. For many genes the fold changes are slightly shifted down, i.e. there are fewer bound ribosomes in the stressed sample. However, the effect is only modest and as Figure 5.4 (b) and (c) show does not change the results of the simulation much.

The effects of protein aggregation are more pronounced. Figure 5.5 (a) shows the comparison of the fold changes in the soluble fraction (supernatant) and in total. The



(a) supernatant vs total (b) simulation total (c) simulation supernatant

Figure 5.5: Influence of protein aggregation on the simulation using equilibrium protein synthesis. (a) Comparison of fold changes in the soluble fraction (supernatant) and of all proteins (soluble and insoluble = total). Results of the simulation for total protein (b) and soluble protein (c). The results for the total protein are slightly better.

majority of the proteins that deviate between the two measurements show increased fold changes in the total fraction, consistent with increased aggregation upon heat shock. The results of the simulation are also slightly better for the total protein measurement, but the difference between the two simulation results is only modest. Note that the fractionated proteome measurements for the total fraction are only available for 30 min so that the measurements after 10 min had to be inferred. Also the number of identified proteins was lower compared to the corresponding normal proteome measurements (1,635 identified by the fractionated measurement compared to 2,382 identified by the standard measurement). This impedes the simulation using the fractionated data and makes the simulations using the different kinds of proteome data incomparable. We will thus focus on the standard data even though we miss the effect of aggregating proteins.

Comparison of Fitting Methods

As comparison, we first want to demonstrate the difference of fitting methods using the data of Lee et al. They analyzed yeast salt stress for which no ribosome profiling but only gene expression data was available. Thus, they reasoned that it is necessary to fit the protein synthesis parameter k_s for each protein separately, to model differences in the translation efficiency. Furthermore, because cell-division is arrested for up to 45 min after the stress is applied, they assumed that protein synthesis varies before and after the arrest and thus fitted two separate synthesis parameters k_s for the one time point before 45 (30 min) and the remaining time points.

Figure 5.6 shows how the simulation results change when the synthesis parameter k_s is fitted in different ways. Given that the ribosome profiling data correlates well to the expression data in heat shock, one could assume that the same is true in salt stress. Then, the k_s parameter could be calculated from the equilibrium/unstressed state and remain

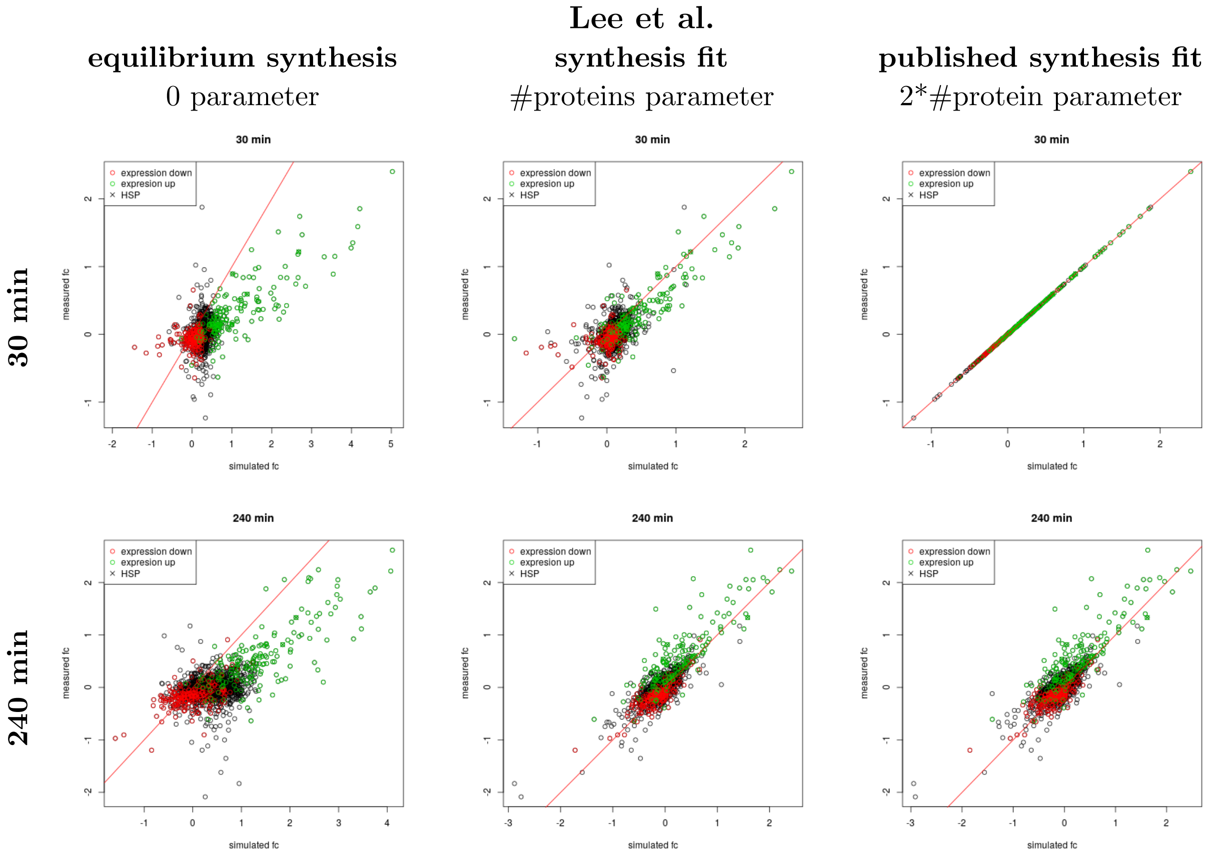


Figure 5.6: Comparison of simulated and measured protein fold changes for the Lee et al. data. Of the 5 available time points only the first (30 min, first row of plots) and the last (240 min, second row of plots) time points are shown. In the first column the protein synthesis is calculated from the equilibrium measurements (no stress), in the second column one parameter is fitted for each protein and applied to all time points and in the last column two parameters are fitted for each protein, one for the first time point and one for the remaining time points.

unchanged over time. The left column of Figure 5.6 shows how this assumptions affects the simulation. Most upregulated proteins are expected to show higher fold changes than those measured over the complete course of the experiment. Also, there are some proteins that are expected not to change at all, but are regulated in the real measurements. The second column shows the effect of fitting one k_s parameter per protein for the complete time series. Here, the latter time points are predicted well, but the first time point exhibits some outliers. The last column shows the results for the model as published, with two fitted parameters per protein. The latter time points are very similar to the fit using only one k_s parameter per protein. For the first time point, there is now one parameter fitted to one measurement, so the fit is perfect and completely uninformative. As there was only one measurement before the cell-division arrest it remains unclear whether the fitted synthesis rate during the arrest would be able to predict additional protein measurements during this period. Without ribosome profiling data and additional time points during the arrest phase, one cannot decide which mechanisms cause the differences.

In comparison to the data used by Lee et al. our heat shock data contains (only) two early time points and ribosome profiling data. Using this data we test four different fitting methods to predict the protein changes:

(a) **equilibrium:** no fitting, the protein degradation is taken from the measured protein half-lives and the synthesis rate is calculated from the measurements at equilibrium/no stress,

(b) **degradation group:** the proteins are grouped by their changes in the ribosome profiling data (up, down and unchanged) and for both the up- and downregulated groups one factor for the measured protein half-lives is fitted and applied to all proteins in this group, while the unchanged proteins are simulated as in the equilibrium fit

(c) **degradation fit:** for each protein a factor is fitted for the measured protein half-lives and applied to all time points and

(d) **synthesis fit:** the protein synthesis rate is fitted for each protein separately.

Figure 5.7 shows the results of these methods. In the first column the protein synthesis rate is calculated from the **equilibrium** measurements and the measured protein half-lives were used. Here for the 10 min measurement, most unchanged proteins and also some upregulated proteins are predicted correctly, but there are many outliers that show less pronounced fold changes than expected. This trend becomes even more obvious in the 30 min measurements, where also many unchanged proteins are expected to show more extreme changes.

In the second column the proteins are grouped by their translation changes (*RP up*, *RP down* and *RP unchanged*) and for each of these groups different degradation factors are assumed (**degradation groups**). The underlying rational is that some proteins become less stable, some are not affected and some are becoming more stable, e.g. because they are protected by chaperones. The changes in the translation correspond to the cell's reaction to these modified protein stabilities in order to maintain homeostasis. Thus we can use the changes in the translation to define the groups of proteins whose stability is affected similarly by heat. The measured protein half-lives are then modified by a factor that was fitted for the corresponding group to take the altered stability into account. We fit one factor each for the proteins with up- and downregulated translation and use the measured half lives for the unchanged proteins. For the *RP up* proteins modifying the half lives by a factor of 0.57 yielded the minimal fold change deviations, while for the *RP down* a factor of 1.69 was optimal. This corresponds to increased degradation for the upregulated proteins and decreased degradation for the downregulated proteins. This fit yielded fewer outliers in the comparison of the measured and simulated fold changes and especially after 30 min the performance of the fit improved.

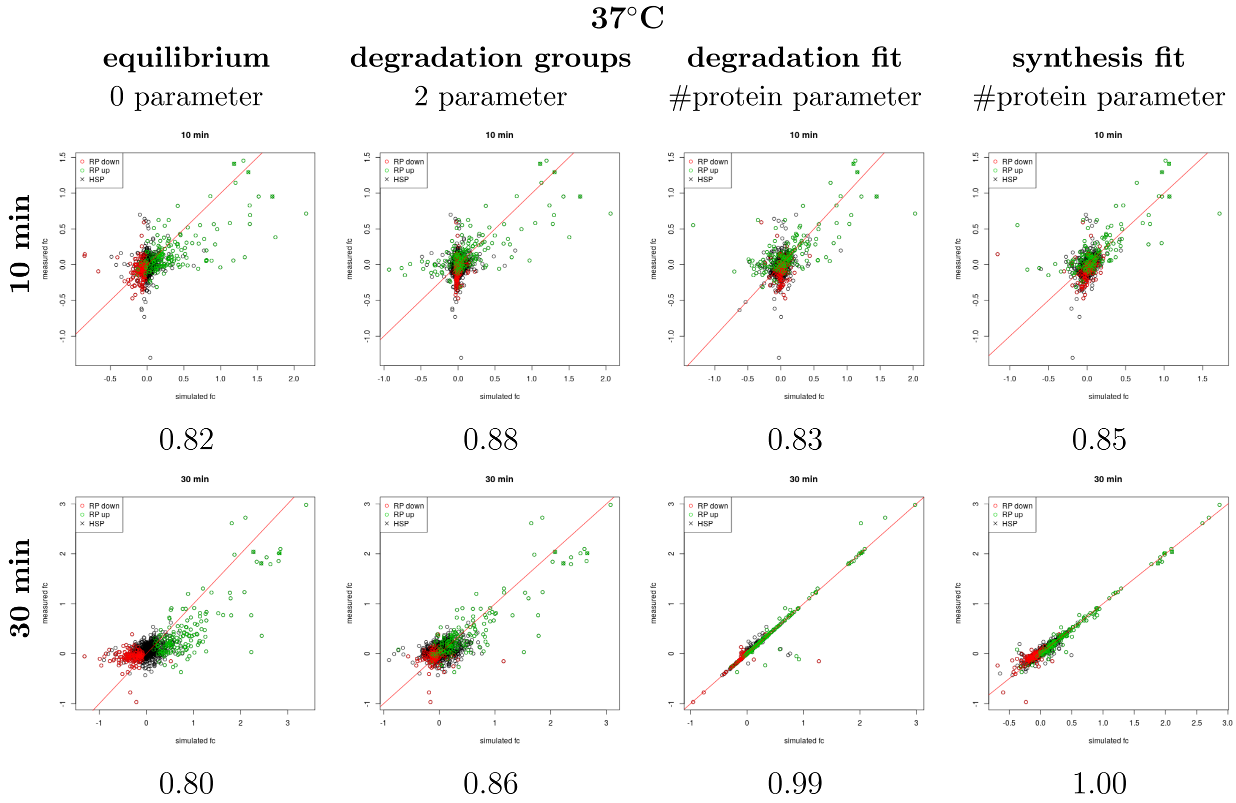


Figure 5.7: Comparison of the simulated and measured protein fold changes at 37°C using different fitting methods. Below each plot the percentage of noise-consistent proteins (see Table 5.1) is given. The degradation and synthesis fits nearly perfectly reproduce the measured fold changes, possibly due to overfitting. The proteins that are up- and downregulated in the ribosome profiling (RP) data are highlighted, as well as the heat shock protein (HSPs). All HSPs that could be simulated are near the diagonal for all fits.

For the two fits shown in the last two columns, it is assumed that heat can have different effects on individual proteins. Some proteins are unstable, aggregate and subsequently degrade much quicker than under equilibrium conditions, or their protein synthesis is affected by ribosome stalling or similar mechanisms. The third column shows the results for the **degradation fit** when the protein decay rate is fitted, while in the fourth columns the protein synthesis rate is optimized by the **synthesis fit**. As protein decay follows an exponential function while protein synthesis grows linearly the two fits are not equivalent. For both fits the changes after 30 min could be simulated correctly for most proteins, while at 10 min there are more outliers. Overall, both fits are quite similar so that one cannot decide whether an alteration of protein synthesis or decay is the main factor that leads to the differences between the simulated and measured changes under equilibrium conditions.

Figure 5.8 shows the comparison of simulated and measured protein fold changes at 42°C. Overall, for all fitting variants the measured fold changes cannot be reproduced as good as for 37°C. The most striking outliers are a group of proteins that are strongly

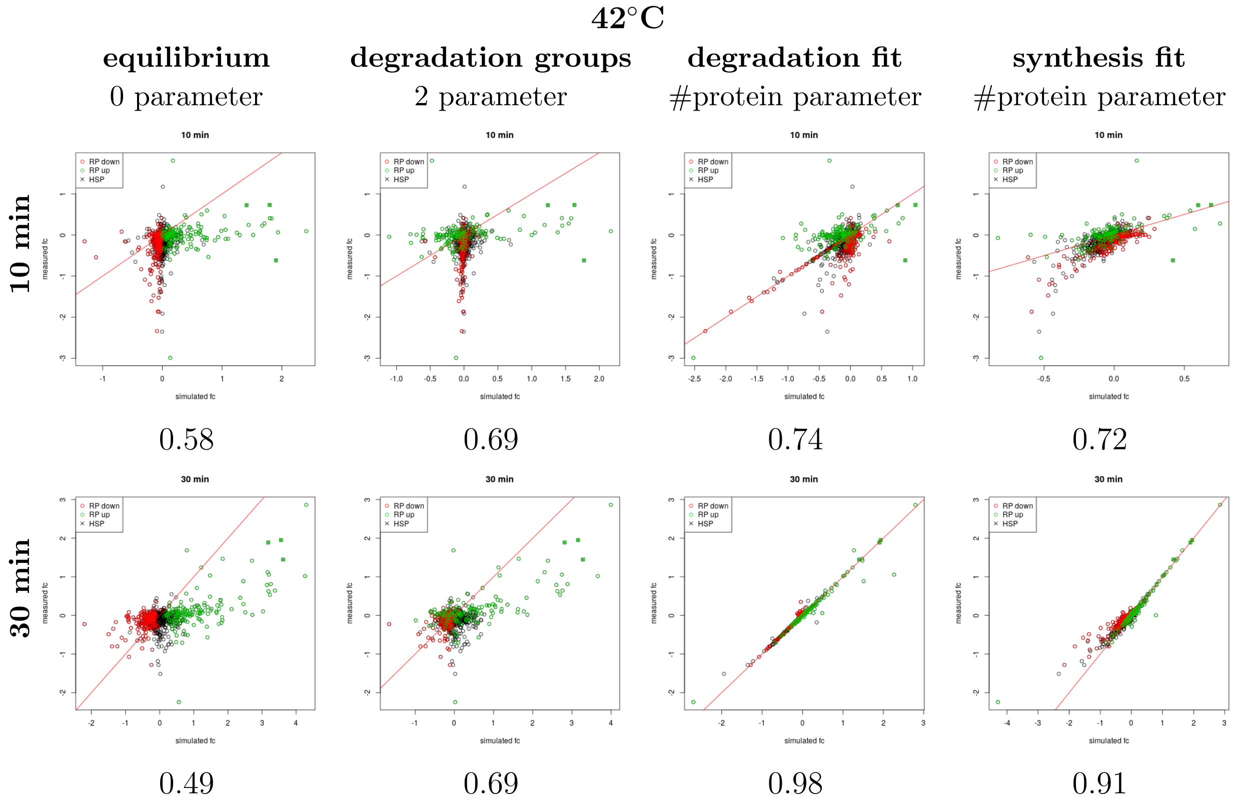


Figure 5.8: Comparison of the simulated and measured protein fold changes at 42°C using different fitting methods. Below each plot the percentage of noise-consistent proteins (see Table 5.1) is given. For all fitting variants the fold changes could be reproduced inferior to 37°C. For a group of proteins the downregulation is more pronounced after 10 min compared to 30 min, while their translation does not show a similar pattern. This indicates a time-dependent mechanisms that cannot be modeled with the limited number of measured time points available here.

downregulated after 10 min but whose protein abundance does not decrease further after 30 min, but instead stays the same or even increases. For most of these proteins the translation is not increased between 10 and 30 min, but stays at the same level or even decreases. Thus, as the direction of the changes on translation and protein level contradict each other, these protein changes cannot be explained by synthesis and decay rates that are constant over time. To model such a time-dependent mechanisms additional measurements at additional time points are necessary.

For the changes after 30 min of heat shock two independent proteome measurements are available that can be used to evaluate the fitted parameters. Figure 5.9 shows the comparison of simulated and measured changes when the parameters that were fitted for the proteome time series are applied to two independent proteome measurements after 30 min. The comparison of the different types of fits show that the **degradation fit** and **synthesis fit** that both use individual parameters for each protein yield many more outliers in the independent test sets, indicating overfitting.

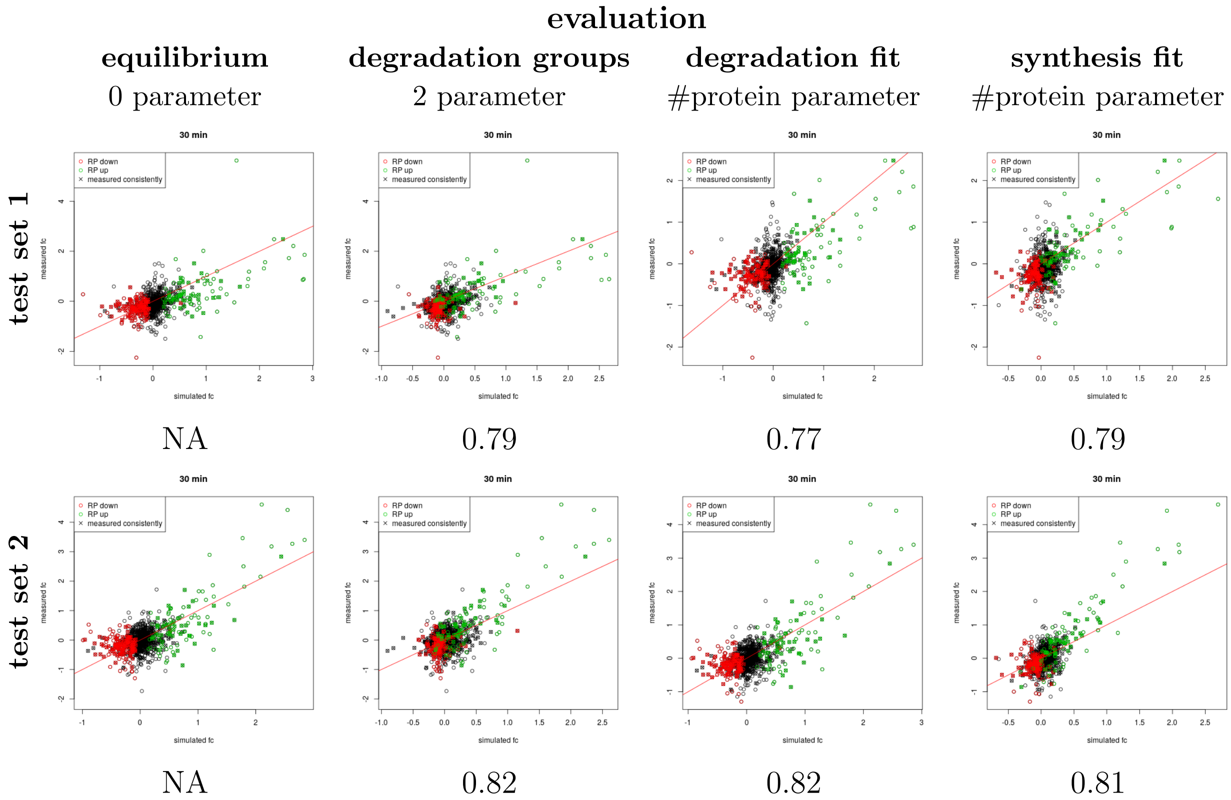


Figure 5.9: Comparison of the simulated and measured protein fold changes when the fitted parameters are applied to independent proteome datasets. Below each plot the percentage of noise-consistent proteins (see Table 5.1) is given. Here we additionally highlight the proteins that were consistently measured (i.e. fold change difference below 0.5) between the two test set measurements. Most proteins that deviate from the diagonal are not measured consistently and might be errorneous measurements.

Moreover, we analyzed systematically for each dataset and fitting method how many proteins could be simulated within their error margin. For this we determined for each protein the interval of simulation results when different replicates of the input data were used. If this range of simulated protein abundances overlaps with the interval of the replicates of the proteome measurement, the protein is correctly predicted given the measurement noise. Table 5.1 shows the fraction of correctly predicted proteins for each combination of fitting method and input datasets for both 37°C and 42°C. In general, the results of most fitting methods perform better on the 37°C data. In the fractionated proteome data, the total fraction does not perform better than the soluble fraction, as would be expected if unmeasured aggregated proteins contribute to the inconsistency between ribosome profiling and proteome measurements. The **degradation groups** fit showed a clear improvement over using the equilibrium simulation. For the normal setup 76% of all proteins are simulated within their error margin at 37°C. For the **synthesis fit** and the **degradation fit** this number increase only moderately to 84% and 81% respectively, even though many more parameters are used. Given these results together with the results of the independent measurements, the **degradation groups** fit is the most parsimonious fit that yields sufficiently good results.

5.5 Discussion

The expression of hundreds of genes are up- or downregulated in the heat shock response in yeast. On the protein level, however, there are far fewer and less pronounced changes. This inconsistency could be explained by a change in translation, but ribosome profiling experiments showed that the ribosome-bound mRNAs exhibit similar changes as the total mRNA. However, the ribosomes are not distributed uniformly over the transcript, but are enriched in the beginning of the transcript upon stress, so that the ribosomes might be stalled and less protein is produced.

An alternative explanation for the discrepancies between protein and mRNA levels are the different abundances and half-lives of proteins and mRNAs. To take these factors into account a quantitative modeling using equilibrium synthesis and decay rates was done. This showed that especially for many of the changing genes the simulated protein abundances are inconsistent with the measured abundances. Even when noise of the measurements is taken into account only 67% of the proteins could be simulated correctly at 37°C and at 42°C this fraction is even lower at 34%.

Moreover, we used various fitting variants to analyze different hypotheses for the cause of the deviations. Overall we tested four different hypotheses: (a) there is no change compared to equilibrium conditions, (b) the proteins with up- and downregulated translation are affected differently by the heat: the degradation of the upregulated proteins is increased while it is decreased for the downregulated proteins, (c) the degradation changes for each protein individually and (d) the synthesis rate changes for each protein individually.

Using these variants the changes in protein abundances can be simulated and compared to the changes observed in the time series measurement as well as to independent measurements after 30 min. This showed that the fits assuming an independent parameter per protein (either for synthesis or decay) are prone to overfitting. Including different degradation rates for the proteins that are up- and downregulated in the ribosome profiling data notably improved the results compared to the simulation using equilibrium parameters. It is thus the most parsimonious fit as it involves only 2 parameters and provides robust good results. The rationale behind this fit is that some proteins are degraded faster when the temperature is increased, and in order to maintain homeostasis the translation of these proteins is increased to compensate for the increased degradation. Another group of proteins become more stable upon heat, so that the cell can decrease their translation to maintain homeostasis and more ribosomes are available for the increased translation of the proteins with increased degradation.

Unfortunately, our measurements only cover two time points with increased temperatures (10 and 30 min), which limits the usefulness of the simulation. Additional measurements at further time points would increase the confidence in the conclusions taken from the simulation and it could be possible to also model a time-dependent mechanism to explain the effects observed at 42°C.

applied to		**trainings set**						**test set**			
		37°C			**42°C**			**37°C**		**42°C**	
fit data		**t10**	**t30**	**all**	**t10**	**t30**	**all**	**t30**	**t30**	**t30**	**t30**
equilibrium	normal	0.82	0.80	0.67	0.58	0.49	0.34	NA	NA	NA	NA
	stalled	0.83	0.83	0.69	0.60	0.53	0.36	NA	NA	NA	NA
	fract. soluble	0.78	0.85	0.77	0.75	0.76	0.69	NA	NA	NA	NA
	fract. total	0.78	0.83	0.73	0.64	0.71	0.59	NA	NA	NA	NA
	fract. total+stalled	0.79	0.84	0.75	0.66	0.65	0.55	NA	NA	NA	NA
degr. groups	normal	0.88	0.86	0.76	0.69	0.69	0.54	0.79	0.82	0.73	0.63
	stalled	0.88	0.89	0.79	0.68	0.70	0.53	0.80	0.82	0.71	0.64
	fract. soluble	0.85	0.85	0.80	0.76	0.78	0.72	NA	0.83	NA	0.64
	fract. total	0.83	0.87	0.79	0.67	0.73	0.61	NA	0.81	NA	0.63
	fract. total+stalled	0.84	0.86	0.78	0.66	0.72	0.60	NA	0.81	NA	0.64
degrad. fit	normal	0.83	0.99	0.81	0.74	0.98	0.67	0.77	0.82	0.72	0.65
	stalled	0.82	0.99	0.80	0.73	0.98	0.67	0.77	0.81	0.73	0.66
	fract. soluble	0.89	0.98	0.89	0.89	0.97	0.86	NA	0.73	NA	0.53
	fract. total	0.88	0.99	0.87	0.82	0.97	0.78	NA	0.71	NA	0.50
	fract. total+stalled	0.89	0.99	0.88	0.83	0.97	0.79	NA	0.71	NA	0.51
synthesis fit	normal	0.85	1.00	0.84	0.72	0.91	0.70	0.79	0.81	0.72	0.62
	stalled	0.84	1.00	0.84	0.70	0.91	0.68	0.78	0.81	0.71	0.62
	fract. soluble	0.94	0.99	0.94	0.94	0.99	0.94	NA	0.62	NA	0.47
	fract. total	0.91	1.00	0.91	0.86	1.00	0.86	NA	0.68	NA	0.45
	fract. total+stalled	0.91	0.99	0.91	0.85	1.00	0.85	NA	0.67	NA	0.47

Table 5.1: Overview of simulation results for both 37°C and 42°C. Each cell gives the fraction of proteins whose predicted interval given noise overlaps with the observed interval of the replicates for the two time points and those proteins with overlapping intervals for both time points. The last four columns show the results when the parameters that were fitted for the corresponding dataset are applied to the two independent test sets. As the fractionated measurements are one of the two test sets these results are omitted. Note that different subsets of proteins are available for the different proteome datasets (normal/stalled and fractionated) which makes them incomparable and that the fractionated measurements are only available for 30 min and had to be infered for 10 min.

Chapter 6

YESdb: Interactive Integrated Analysis of Stress Datasets

Motivation

In the last chapter we described a very specific approach for the integrated analysis, that is used to model in detail a specific part of the central dogma of biology using a specific set of measurements. Here, a much more general approach is used to integrate multiple datasets that can be used to analyze various research questions using different kinds of measurements.

Here, we describe a Petri-net based workflow system that uses fundamental operations to define, combine and characterize sets of interesting genes from multiple datasets. This allows to tackle various research questions, such as the differences between different stimuli or which technical biases exist on different experimental platforms.

While several databases such as GEO, SRA or PRIDE exist that contain large collections of publicly available high-throughput datasets, the direct use of such integrative approaches in these large-scale databases is hindered by the need to find and preprocess the available datasets for the given research question. These huge repositories often provide both raw and processed data, but not the differential data that is most suitable for an integrative analysis.

YESdb is a database that contains preprocessed differential datasets of the yeast stress response. The datasets are annotated with the kind and strength of the applied stress, the strain and experimental technique that were used and the time at which the measurement was taken as well as the publication date. A web interface allows to quickly find relevant datasets that match a given combination of these annotations and analyze them using the workflow system.

The results of each step in such a workflow can be visualized in an interactive report that can also contain workflow independent visualization that e.g. characterize the selected datasets. This way, comprehensive reports can be created that can also be saved and shared.

Publication

The content of this chapter is submitted to Database [10]. Here the manuscript is reformatted.

Author Contributions

Evi Berchtold designed and implemented the database, performed the analysis and wrote the paper. Gergely Csaba searched the meta-data of GEO and SRA to find the relevant datasets and preprocessed the RNAseq data. Ralf Zimmer supervised the project and edited the paper.

Availability

YESdb is available at

`https://services.bio.ifi.lmu.de/YESdb`

6.1 Introduction

More and more high-throughput data is made publicly available in databases like GEO [6], SRA [62] or PRIDE [98]. Published data can be used to complement newly measured data in various ways. Meta-analyses integrate diverse datasets from different studies, tissues or species to draw unbiased conclusions. While meta-analyses usually focus on data from the same or similar platforms, another way to benefit from published data is to integrate datasets from the same or a similar condition measured on different platforms (e.g. RNAseq and microarray data). Systematic biases of one platform can thus be identified and corrected for. Similarly, datasets that measure different levels (e.g. expression and protein levels) of the same condition can be combined to obtain a more complete picture of the changes in the cell.

Even though the integration of multiple datasets can improve the analysis many studies ignore published data that could be integrated in their analysis. The first hurdle for integrative analyses is of course to find data that fits, which often involves reading detailed experimental descriptions to uncover how similar the conditions are. Furthermore, integrative analyses are often hindered by the need to preprocess the raw data that is stored in the public databases. Especially when the published data is measured on a different platform, a different preprocessing workflow has to be used.

To facilitate the use of published data some databases offer analysis possibilities directly. GEO introduced the GEO2R tool which allows to use GEO datasets directly in R analyses. This is a very powerful tool but limited to users that are familiar with the R programming language. Other databases such as MEM [2] and SPELL [51] also allow the user to do some analyses directly on their website, but they focus mainly on co-expression studies.

Workflow managers (see [67] for a recent review) enable the user to conduct complicated pipelines to process the data. This allows the user to easily test the influence of parameter settings, or the choice of specific methods. A major limitation for using a workflow manager for an integrative analysis is the search for and import of the already published data. Furthermore, pipelines are typically used for a standard analysis of the data (e.g. to derive the differentially expressed genes in an experiment), as the more specific downstream analyses cannot normally be re-used for another experiment, and the next step can often not be defined in advance, as it depends on the results of the previous step.

The stress response in Saccharomyces cerevisiae is an especially well studied system for which many different datasets are available. However, there are still many unsolved questions of how the system is regulated for the different kinds of stress. To study the conserved and divergent parts of the system an integrative analysis is needed.

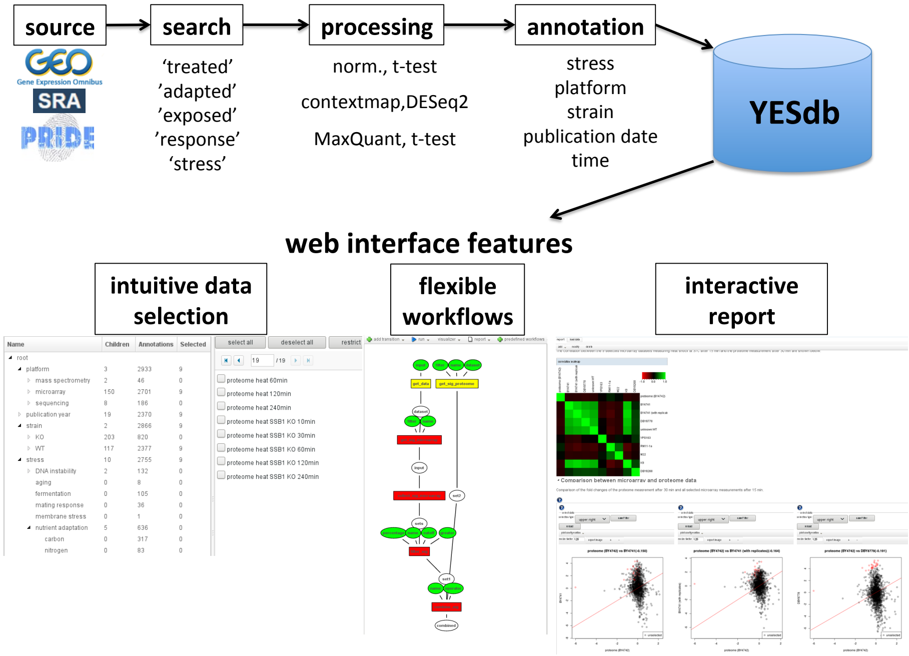
yStreX [100] collected, classified and preprocessed several datasets measuring different stress conditions in yeast. It allows to identify differentially expressed genes, to find conditions in which a gene is differentially expressed and enrichment analyses for single and multiple conditions. However, it has also several limitations: the collected datasets are required to have more than two replicates, so that many time series analyzing different kinds of stress with one replicate per time point are missing. Furthermore, it contains only gene expression data measured by microarrays, so that proteomics or sequencing datasets are not contained. This results in a total of 121 conditions, which is only a small subset of the available data.

Here, we describe YESdb a database that contains preprocessed differential expression data for various types of stress in the model organism Saccharomyces cerevisiae. To make best use of the data, the database contains a Petri net-based workflow system, that allows the user to integrate multiple datasets. The results of the workflow are visualized in interactive reports, that contain a visual summary of each step in the workflow. Several runs of a workflow with different parameters can be directly compared in these reports. This way, the impacts of individual parameters in a complex analysis can easily be analyzed.

6.2 Data

6.2.1 Data search strategy

To find the relevant datasets, the meta-data from GEO was filtered for datasets measuring RNA in Saccharomyces cerevisiae, and the resulting datasets were searched for ’treatment’/’treated’, ’adaptation’/’adapted’, ’exposure’/’exposed’, ’response’ and ’stress’. This yielded 386 GEO Series of which most were microarray datasets contained in GEO and only 35 corresponded to high-throughput sequencing datasets contained in SRA. For proteomics data, there are far fewer datasets available in PRIDE, which are unfortunately less standardized and less comprehensively annotated. Therefore, we manually selected the relevant datasets for which MaxQuant [22] output was available and for which the individual conditions could be identified in the output.



Name	Children	Annotations	Selected
root			
platform	3	2933	9
mass spectrometry	2	46	0
microarray	150	2701	9
sequencing	8	186	0
publication year	19	2370	9
strain	2	2866	9
KO	203	820	0
WT	117	2377	9
stress	10	2755	9
DNA instability	2	132	0
aging	0	8	0
fermentation	0	105	0
mating response	0	36	0
membrane stress	0	1	0
nutrient adaptation	5	636	0
carbon	0	317	0
nitrogen	0	83	0

19 /19

☐ proteome heat 60min
☐ proteome heat 120min
☐ proteome heat 240min
☐ proteome heat SSB1 KO 10min
☐ proteome heat SSB1 KO 30min
☐ proteome heat SSB1 KO 60min
☐ proteome heat SSB1 KO 120min
☐ proteome heat SSB1 KO 240min

Figure 6.1: Overview of the data sources and web interface features of YESdb. All stress response datasets from GEO, SRA and PRIDE are selected, processed to differential conditions (DC) and annotated. The web interface features an intuitive data selection interface, workflows that allow to execute complex analyses and interactive reports to summarize and visualize the results.

6.2.2 Data processing

YESdb contains already differential conditions (DC), so that the user does not have to identify replicates and the conditions that should be compared. To construct this configuration we used a semi-automatic framework that first automatically identifies replicates and control/unstressed conditions which are then manually corrected and completed. Additionally, time or concentration courses are saved as *series*, i.e. lists of DCs together with the corresponding time or concentration.

For GEO the datasets are already processed in most cases. A simple median normalization is used to make the individual samples comparable while not distorting the already processed (and typically normalized) values too much. The configurations are then used to calculate $\log_2$ fold changes and t-test p-values where applicable (i.e. raw measurements and replicates are available). The SRA datasets are mapped by ContextMap [12] and differential expression was analyzed by DESeq [70]. For PRIDE SILAC fold changes of replicates are combined by taking the mean and $\log_2$ fold changes and t-test p-values are calculated for LFQ data. In all cases, replicates and raw measurements are saved if available so that they can be used for visualization and filtering.

annotation	**DC**	**series**	**children**
platform	2933	392	166
microarray	2701	356	153
sequencing	186	29	8
mass spectrometry	46	7	2
publication year	2701	298	19
strain	2866	380	322
wild type	2377	308	117
knock-out	820	113	203
stress	2755	378	84
other	860	102	14
nutrient adaptation	636	90	5
oxidative stress	460	58	20
osmotic stress	361	56	19
temperature	278	47	14
DNA instability	132	25	2
fermentation	105	20	-
mating response	36	6	-
time	2072	-	124
30 min	283	-	-
60 min	109	-	-
2 h	141	-	-
20 min	92	-	-
15 min	72	-	-
10 min	65	-	-
5 min	57	-	-
...			

Table 6.1: Overview of the annotations of the datasets contained in YESdb. Only the first level of annotation is shown, most annotations contain additional levels such as the specific platform or strain used or the strength of the applied stress. For the time annotation only the most frequent entries are shown. The number of all (also indirect) child annotation terms are given in the last column. The data is processed to differential conditions (*DC*) and (time or concentration) *series*.

6.2.3 Data annotation

We created an ontology of annotations to make it easy to find the relevant datasets for a specific analysis. This ontology contains the experimental platform, the publication date, the yeast strain that was used (including which genes were knocked-out), and the kind of stress that was applied. Each GEO/SRA/PRIDE dataset was manually mapped to all relevant terms in this ontology. To select the relevant datasets, we provide an easy to use interface where the ontology can be browsed and the *DCs* or *series* annotated to a selected term are shown. These entries can then be selected or excluded individually or all at once, and the entries selected so far can be restricted to those annotated in the currently selected ontology term. This allows the user e.g. to select all *series* annotated to heat shock at 37°C and 39°C, and restrict this selection to those *series* that were measured by microarray and exclude all *series* that used knock-out strains.

Table 6.1 shows the first levels of this annotation hierarchy. The database contains 2933 *DCs* and 392 *series*. Of these 820 *DCs* measure 203 different knock-out strains and 2.377 *DCs* measure 117 different wild type strains. Oxidative stress, osmotic stress, carbon source adaptation and temperature adaptation are the best studied kinds of stress in our database, containing between 278 and 460 *DCs*.

6.3 Workflows

We implemented a Petri net-based workflow system to allow the user to easily perform integrative analyses of the datasets in the database. This system facilitates the identification of interesting genes from several datasets and to combine them in a flexible way to analyze different hypotheses. Table 6.2 shows an overview of the available transitions. There are transitions to define and combine sets of entities, for downstream analyses such as enrichment or simple network analysis and helper transitions to e.g. modify the differential conditions.

These transitions can be connected to elaborated workflows. Figure 6.2 shows an example workflow. It consists of multiple transitions that can also depend on each other, i.e. the output of one transition is used as input for another transition. These workflows can be executed automatically, or single transitions are selected for execution. Executing single transitions allows to interactively evaluate the results of the transition and modifying the inputs if necessary, before executing the subsequent steps from the workflow.

The tokens in the workflow system can have several types: *DC*, *series*, *set*, *DAG* and *network* as well as the simple types *string*, *boolean* and *number*. To allow for transitions that have a variable number of inputs or outputs of the same type (e.g. to calculate the intersection of the differential genes from several *DCs*), we introduce the notion of token lists, which are simply lists of tokens of the same type. There are helper transitions to combine several tokens to a list token or to isolate the individual tokens from a list token.

The initial tokens can be extracted from the *DC* and (time/concentration) *series* contained in the database. Additionally, the *DAGs* and the corresponding *sets* of the gene ontology [41] and different kinds of *networks* for yeast, such as YEASTRACT [93], BioGRID [21], post-translational modification networks [79, 31, 29] and manually curated stress networks [59] are available.

Name	**Description**
Set from DAG	loads a list of *sets* from a *DAG*, e.g. GO
Set from DiffCond	defines a *set* from a *DC* by filtering the measurements (fold change, raw or p-value)
Binary Set Combination	combines two *sets* by set operations (intersect, difference and union)
Multi Set Combination	combines multiple *sets* by set operations
Count Filter	defines a *set* of the genes that are contained at least/most a given number of times in a list of *sets*
Enrichment	calculates enrichment of a *set* in a list of *sets*
Subnetwork	extracts the subnetwork of a *set* from a *network*
Reverse fold change	swaps case and control conditions of a *DC*
Dataset fold change	generates a new *DC* that is the fold change between two *DCs* (e.g. DC1=stress1 vs control, DC2=stress2 vs control → DC1 vs DC2=stress1 vs stress2)
Series2DiffCond	extracts all *DCs* from a *series*
Collector	combines several tokens of the same type to a list
Distributor	extract the individual tokens from a list

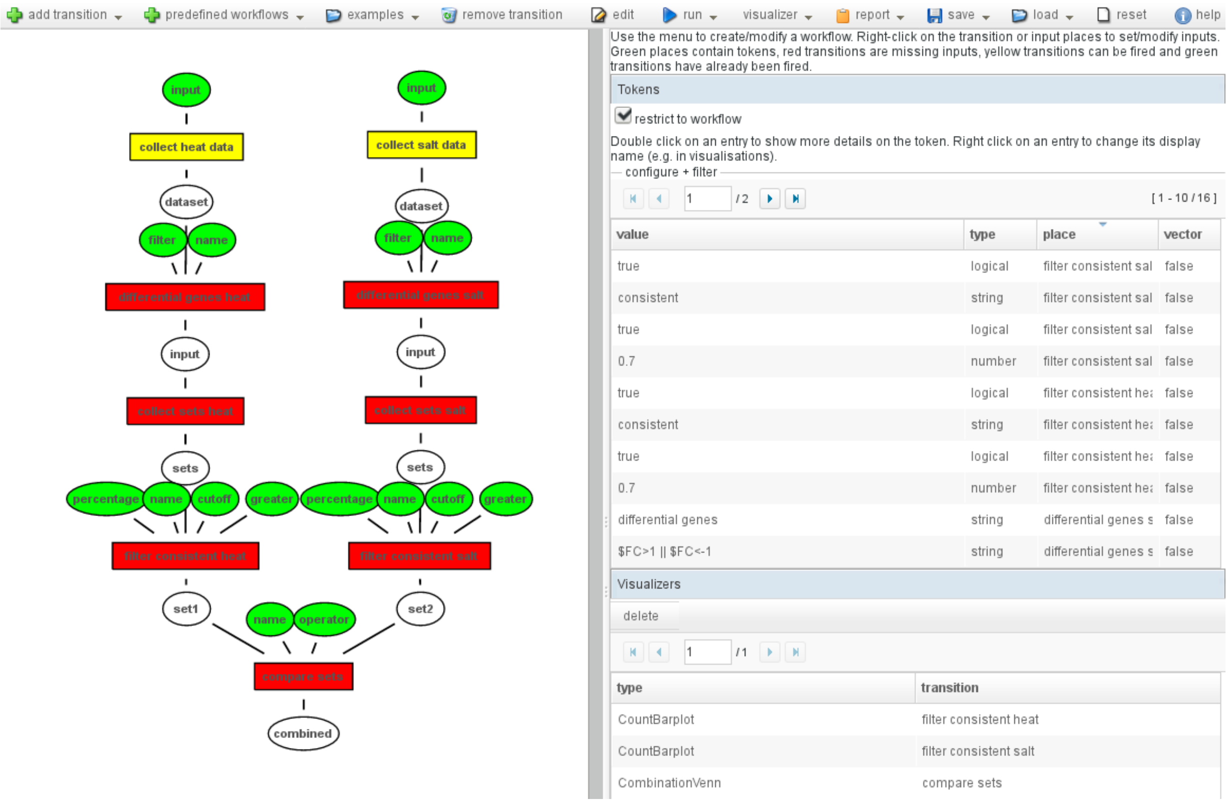
Table 6.2: Overview of workflow transitions. The first block of transitions defines *sets* of interesting genes, the second block characterizes *sets* of genes and the last block contains helper transitions.

6.4 Interactive report

The result of a workflow is not only the final output, but intermediate results can be just as interesting. To provide a convenient way to get an overview of all the results, the user can add to each transition one or more visualizers. When the workflow is executed, the visualizers generate plots, tables or network views, that are all added to one report (see Figure 6.3). For most transitions there are standard visualizers, but additionally, the user can also define custom visualizations to be included in the report, by defining the plot type and inputs. Most visualizations are interactive, so that the associated data of points in a plot or rows in a table can be retrieved. In the example in Figure 6.3 the genes selected in the left scatterplot comparing oxidative stress and heat shock are not only listed below the plot but also highlighted in the right scatterplot comparing oxidative and osmotic stress.

The resulting report can be edited, by adding and removing sections, visualizations, and descriptive text or changing the order of the elements. This way a report that summarizes the results of the workflow is created. It can then be saved as an xml file, which can be uploaded to our website to show the report. This allows to share the results with collaborators or to save intermediate results for later refinement.

If the workflow is executed again with different inputs, another report with the same visualizations using the new data is created. If the two runs of the workflow should be compared, a joined report that contains the results for both runs next to each other is



Use the menu to create/modify a workflow. Right-click on the transition or input places to set/modify inputs. Green places contain tokens, red transitions are missing inputs, yellow transitions can be fired and green transitions have already been fired.

Tokens

☑ restrict to workflow

Double click on an entry to show more details on the token. Right click on an entry to change its display name (e.g. in visualisations).

configure + filter

1 /2 [1 - 10 / 16]

value	type	place	vector
true	logical	filter consistent sal	false
consistent	string	filter consistent sal	false
true	logical	filter consistent sal	false
0.7	number	filter consistent sal	false
true	logical	filter consistent he	false
consistent	string	filter consistent he	false
true	logical	filter consistent he	false
0.7	number	filter consistent he	false
differential genes	string	differential genes s	false
$FC>1 || $FC<-1	string	differential genes s	false

Visualizers

delete

1 /1

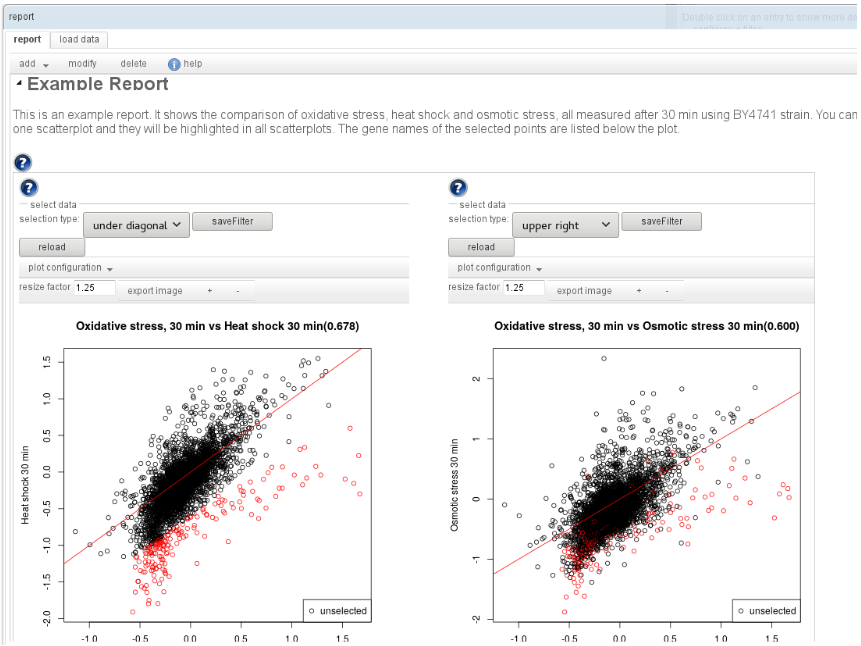
type	transition
CountBarplot	filter consistent heat
CountBarplot	filter consistent salt
CombinationVenn	compare sets

Figure 6.2: Example workflow. On the left the Petri net workflow comparing the consistent genes between heat and salt stress is shown. On the right an overview of all tokens used in the workflow is shown. The color of the inputs (ellipses) indicate whether it contains a token (green) and the color of the transitions (boxes) show whether it has been fired (green), can be fired (yellow) or cannot be fired because input tokens are missing (red).

produced. This allows the user to easily compare different parametrizations of the same workflow, e.g. to compare the effects of different cutoffs for the definition of differential genes or analyzing another type of stress.

6.5 Example analysis

Here we present an example analysis that compares the genes involved in two types of stress: heat shock and osmotic stress. Already in 2000, Gasch et al. [37] showed that yeast responds similarly to a wide range of different types of stress including heat shock and osmotic stress. They observed up- and downregulation of two clusters of genes which they termed 'environmental stress response'. Moreover, the survival of one type of stress can 'cross-protect' yeast cells from a different type of stress, as e.g. heat shock increases tolerance for osmotic stress [75]. To analyze which genes are unique to the two types of stress and which are shared, we first identify the genes that are consistently changed for each type of stress, and then these two sets are compared to each other.



Example Report

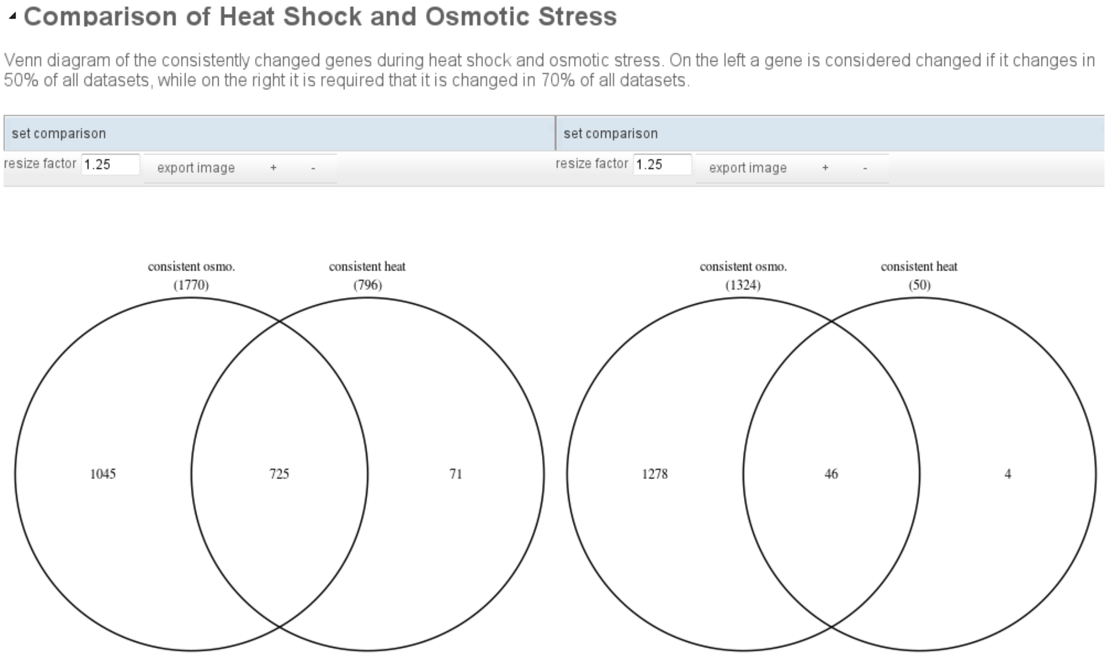
This is an example report. It shows the comparison of oxidative stress, heat shock and osmotic stress, all measured after 30 min using BY4741 strain. You can
one scatterplot and they will be highlighted in all scatterplots. The gene names of the selected points are listed below the plot.

Figure 6.3: An example report. A report consists of sections, with text and interactive visualizations. An editor allows to modify/add/delete the individual elements and to change their order. Many of the available visualizations are also interactive. E.g. in the scatterplot shown here, sets of genes can be selected to highlight them (also in other subplots of the visualization) and to display the labels of the selected points in a list below the plot.

YESdb already contains a predefined workflow to define the consistently changed genes from a list of datasets which can be added to an analysis. This predefined workflow selects for each of the datasets the changed genes and uses the 'CountFilter' transition to identify the genes that are changed in a given fraction (e.g. 70%) of all datasets. We call the resulting set consistently changed genesto capture the 'core' stress response and not technical noise or bias. In the predefined workflows contained in YESdb most inputs are set to appropriate default values, and only the datasets that should be analyzed have to be selected.

Using our interface, we can select all heat shock datasets measured at 37°C, exclude all datasets using knock-out strains and restrict the selection to those datasets measured after 15 min, resulting in 19 datasets. Similarly we can select those 10 osmotic stress datasets measured 30 min after 0.4M NaCl was added, that did not use knock-out strains. Overall, 5627 and 3488 genes are changing ($|\text{fold change}| > 1$) in at least one of the selected heat shock and osmotic stress datasets, respectively, of which 796 and 1770 are consistently changed in at least half of the selected datasets.

To compare the sets of consistently changed genes that are the result of the two copies of the predefined workflow, we add a 'binary set combination' transition. This transition applies a set operation (intersect, difference or union) to two given sets. Using this transition we can define the set of genes that is unique for heat shock or osmotic stress, or the



Comparison of Heat Shock and Osmotic Stress

Venn diagram of the consistently changed genes during heat shock and osmotic stress. On the left a gene is considered changed if it changes in 50% of all datasets, while on the right it is required that it is changed in 70% of all datasets.

Figure 6.4: Example report comparing two runs of the example analysis. On the left genes that that are changed in at least 50% of all datasets are considered consistently changed, while on the right a more strict cutoff of 70% is applied. The resulting Venn diagrams are shown side by side so that the user can easily assess the different results of the analysis.

set of genes that are shared between the two types of stress. There are 725 shared genes and 71 and 1045 genes unique to heat shock and osmotic stress, respectively. The resulting workflow is shown in Figure 6.2.

The inputs of this workflow can be varied to analyze the robustness of the results. We could e.g. change the fraction of datasets for consistently changed genes from 50% to 70%. This changes the number of consistently changed genes to 59 and 1324 in heat shock and osmotic stress, respectively. This shows that the selected heat shock datasets are less consistent than the osmotic stress datasets, maybe because the heat shock at 37°C is a very mild stress to which the different wild type strains that were analyzed in the datasets do not react similarly. To visualize the results of such an comparative analysis, a report comparing multiple runs of the same workflow can be created. Figure 6.4 shows such a report. Similarly to the normal report it contains headers and descriptive text and visualizations. Visualizations that are automatically created from visualizers added to transitions are shown for all selected runs of the workflow, side by side. This way the different results can easily be analyzed.

The example presented here is only one of many possible analyses. It can easily be extended to e.g. characterize the resulting gene set further by gene set enrichment. Similar analyses can be used to tackle different questions like how different strains react to stress, how stress strength influences stress response or whether there is a platform bias.

6.6 Discussion

Public databases like GEO or SRA contain thousands of datasets that often measure similar experimental conditions. Combining these datasets can yield more robust results and more insight because technical biases and noise can be removed. If different biological entities like proteins and gene expression are measured, the integration could provide a more complete picture of the changes in the cell. Moreover, different experimental conditions can be compared to identify shared mechanisms.

The stress response system in the model organism Saccharomyces cerevisiae is a well studied system that is nevertheless not completely understood. There are measurements for different kinds of stress, different strengths, different time frames and on different experimental platforms. The integration of these datasets can help to understand the exact changes in response to a single stress and shared and divergent mechanisms between different kinds of stress.

The YESdb contains nearly 3000 differential conditions of yeast stress measurements using microarray, next-generation sequencing and proteomics platforms. It combines the yeast stress-related datasets of GEO, SRA and PRIDE and provides access to already preprocessed data on the level of differential conditions. The datasets are annotated to different kinds of stress, publication years, platforms and strains. An easy to use interface is used to select the relevant datasets for further analysis.

A Petri net-based workflow system allows to combine a given set of transitions to elaborated analyses that identify and combine interesting sets of genes and characterize them. Even though these transitions correspond to quite simple operations, the possibility to combine them in any way allows not only to perform standard analyses, but also customized analyses for specialized research questions.

The results of such analyses can be visualized in an interactive report. For most transitions visualizers can be added to the workflow that will automatically add a visualization of the result of the transition to the report. This can be especially useful to compare different runs of the same workflow that differ in some parameter. The resulting report contains the visualizations side by side so that the effect of the changed parameter on the results of the various steps in the workflow can easily be analyzed. Additionally, the user can create own visualizations by selecting plot type and inputs from all available inputs and (intermediate) results. Many of the visualizations are interactive, e.g. tables are sortable or information about individual points in a plot can be shown. To explain the results and structure the report, text and subsections can be added to the report, so that a human-readable report of the analysis can be created. The report can be saved as xml-file and uploaded to our website to show the report, so that reports can be shared e.g. between collaboration partners.

The annotation contained in YESdb provides a valuable resource for systematic analyses. It can be used to systematically analyze differences between platforms, strains, types of stress or the strengths of the applied stress. Furthermore, it contains datasets for 203 knock-out strains, that can be used to compare the effects of the knock-out in different types of stress or to understand the regulatory mechanisms in general.

While the system has been demonstrated for stress response in yeast, we think that also other research questions can benefit from our system. To use the interactive workflows and reports for another biological system, the corresponding datasets have to be identified, processed and annotated to create the underlying database. Additionally, the set of available transitions can be extended to include more complex operations.

Chapter 7

Discussion

The goal of most bioinformatic methods is to build models that help to understand a given biological system. The resulting models can have different types (e.g. diagrammatic or formal) and different *context-levels*. In this thesis two aspects of this tasks are analyzed: model evaluation and improved model building by integrating different data sources.

The *i-score* presented in Chapter 2 provides an easy to understand evaluation measure of bioinformatic methods to predict active TFs. It assesses the target genes whose changes are strictly inconsistent with the predicted activity states of their corresponding TFs. Moreover, when optimized directly it provides a lower bound of the number of target genes that simply cannot be explained given the available expression data and regulatory network. This lower bound is surprisingly high for most experiments and shows that the available networks are not yet complete even in well studied model organisms such as yeast.

In recent years new high-throughput sequencing techniques were developed that allow to identify the regions of the genome where the chromatin is accessible. Both DNase-seq and ATAC-seq [15, 88, 95] identify regions where the DNA is not tightly packed around nucleosomes and, thus, can be bound by additional regulatory factors such as TFs. While these experimental techniques still do not provide a gold standard of which TFs are active in a given biological condition, differential ATAC-seq or DNase-seq data can yield predicted TF binding sites whose accessibility is changing, most likely because the corresponding TF is binding differentially. In contrast to ChIP-seq, ATAC-seq and DNase-seq can be used to predict the binding of *all* TFs, only limited by the knowledge of their binding motif and the quality of the motif matches. When both differential gene expression and differential chromatin accessibility measurements are available this information can be used similarly to the *i-score* to evaluate active TF prediction tools. As a future application of the *i-score*, such chromatin accessibility data could be integrated in an evaluation procedure. One possibility for this would be to filter those edges from the network whose binding sites are inaccessible and to include edges for binding sites with accessibility changes. Moreover, the *i-score* could even be used to look for missing regulations in the gene regulatory network, when the differentially accessible chromatin regions of unexplained target genes are examined further to find additional motif hits or even overrepresented new motifs.

A different evaluation setup was presented in Chapter 3 where the performance of six breast cancer subtype classifiers was evaluated. For this not only an independent cohort was used, but with Fluidigm also a different experimental platform for gene expression measurement. Most classifiers performed well in this setup, showing that they are very robust and that qPCR measurements of Fluidigm can be used as input for these classifiers. The results of our analysis are compiled in an interactive iReport that allows to analyze the underlying data down to the individual patient.

It would be interesting in the future to extend this interactive analysis of different subtype classifiers to other cohorts. There are several large cohorts available for which the expression was measured using microarrays, so that measurements for all genes needed by the subtype classifiers should be available. This way the performance for different cohorts could be compared directly to each other and also additional available data could be integrated. The Cancer Genome Atlas (TCGA) [18] provides not only gene expression data for over 1000 breast cancer tumors, but also information about SNPs, methylation and copy number variations. Integrating all this information in an interactive iReport could help to formulate new hypotheses about the subtypes of breast cancer or improve the existing classifiers.

The second objective of this thesis is to present methods that generate bioinformatic models that help to explain biological systems. RelExplain (Chapter 4) generates diagrammatic models of biological processes by identifying the subnetwork that best explains the given data for a specific biological process. In contrast to other significant area search methods it takes the functional annotation into account and, thus, yields subnetworks that are specific for the given process. This allows to analyze processes in more detail that are hypothesized to be interesting (e.g. because they were identified to be enriched for changing genes). RelExplain also incorporates consistency of regulatory edges in its edge scoring and is superior in extracting manually curated subnetworks from a context independent large scale network.

The edge scoring used by RelExplain could easily be extended to integrate additional types of measurements in the future. Different types of measurements are meaningful for different types of edges depending of the biological entities (genes, proteins, TFs, etc.) that are involved in the relation described by this kind of edge. This can be integrated into the scoring similarly to the consistency scoring for regulatory edges. Another future direction for RelExplain would be to score the resulting explanations for all processes and use this information to improve the ranking of processes given by enrichment methods. The rational behind this approach is that processes that are false positives in the enrichment method probably will yield worse explanations in RelExplain as the contained changing genes are not directly related.

In Chapter 5 we analyzed the use of formal models using mathematical equations to predict the expected changes in protein abundance given the observed changes in translation during the yeast response to heat shock. First, we analyzed the qualitative changes and found that most genes that are changed consistently in the gene expression and ribosome profiling data remain unchanged in the proteome data. To analyze this inconsistency we applied mathematical modeling to also incorporate the effects of protein half lives and absolute abundances. We tested different hypotheses of how the synthesis and/or degradation rates had to change to explain the inconsistencies between proteome and ribosome profiling data. The most parsimonious fit that yielded good, robust results assumes that there are groups of proteins for which the degradation rate is affected similarly. When the temperature rises, some proteins become more unstable and to compensate for their increased degradation, the translation of these proteins is increased. Another group of proteins become more stable (e.g. because they are stabilized by chaperones) and their translation is decreased. These observed huge changes in gene expression and subsequently translation rate mainly ensure protein homeostasis in the changed environment.

A more general approach to integrate different datasets is described in Chapter 6. Here we present a Petri net like workflow system that can be used to define and characterize gene sets across multiple yeast stress response datasets. The corresponding database YESdb contains all yeast stress response datasets from GEO, SRA and PRIDE. These datasets are annotated with the kind, strength and duration of the stress as well as the experimental platform, publication date and the used strain. These annotations can easily be combined to find and select all relevant datasets to be analyzed in the workflow system. The available transitions are basic operations to define and characterize gene sets that can be combined to elaborated workflows. The results of such an analysis can be summarized in a report with interactive plots and tables to share the results.

A future application using the YESdb data would be a comprehensive comparison of the different types of stress or different strengths of the same stress. As the data is extensively annotated the available datasets for such an comparison can easily be selected. Also the influence of different confounding factors such as the experimental technique or the used strain can be systematically analyzed.

While the chapters in this thesis cover methods to understand quite diverse biological systems with different levels of detail, there are some common themes that proved to be important for many different methods.

When complex biological systems are analyzed the models are often complex themselves and challenging to visualize. Either the level of detail that is shown in the visualization is reduced to only show the overall picture, or only some aspect or part of the system is visualized in detail. An interactive visualization can provide both by linking parts of the overall picture with the detailed visualization. This allows to browse the overall results down to the raw data that was used to generate them and, thus, provides a more comprehensive understanding. The iReport that is available for the evaluation results of the breast cancer subtype classifiers is an example of such an interactive visualization. The user can e.g. select subsets of patients from a result table and analyze them in more detail.

Another application of interactive visualization is to explore alternative solutions. Many methods use some optimization to provide the best solution. They do however only seldom report suboptimal solutions even though they are often only slightly worse than the optimum. When the goal is to understand the model itself it is important to understand which parts of the model are essential for its quality and which can be altered without much loss of quality. For both the *i-score* and RelExplain such an interactive exploration of the solution space is available. For the *i-score* the user can interactively add and remove active TFs to the solution and analyze the effect on the *i-score*. RelExplain allows to analyze alternative paths between genes contained in the optimal solution that are only worse by the predefined margin. This way also the importance of genes that are not differential or not contained in the analyzed process can be assessed.

The other overall principle is that the integration of all available data is needed for a good model. To provide a comprehensive, robust explanation of the system different types of data, measuring different aspects of the system should be integrated. The methods presented in this thesis integrate different types of data: the *i-score* and RelExplain integrate expression data with regulatory networks, as well as functional annotation in case of RelExplain. For the evaluation of the breast cancer subtype classifiers over 700 gene expression measurements and the corresponding clinical data are integrated, and in YESdb nearly 3000 differential conditions measuring some kind of stress response in yeast are integrated. The analysis of the heat shock response used three different types of data: gene expression, ribosome profiling and proteome. For the quantitative modeling additionally protein half lives and total abundances were integrated. The necessity of so many different types of measurements hinders of course the applicability to different systems. But when this data is available it provides a very detailed view on the changes in the system.

In conclusion, depending on the biological problem and the available data at hand different levels of detail can be incorporated in the specific bioinformatic model. Models integrating many different datasets can provide a deeper understanding of the biological system as they e.g. provide quantitative predictions, but the needed input data limits their application to other biological systems. On the other hand, models that can be applied more widely often provide limited details. Thus, for each biological problem the model has to be tailored to include all available relevant data but not require additional data.

Bibliography

[1] R. L. Ackoff. From data to wisdom. *Journal of Applied Systems Analysis*, 16:3–9, 1989.

[2] P. Adler, R. Kolde, A. Tkachenko, H. Peterson, J. Reimand, and J. Vilo. Mining for coexpression across hundreds of datasets using novel rank aggregation and visualization methods. *Genome Biology*, 10(12):R139, Dec 2009.

[3] E. M. Awad and H. M. Ghaziri. *Knowledge Management*. Pearson Education, 2004.

[4] T. L. Bailey, M. Boden, F. A. Buske, M. Frith, C. E. Grant, L. Clementi, J. Ren, W. W. Li, and W. S. Noble. MEME SUITE: tools for motif discovery and searching. *Nucleic Acids Research*, 33((Web Server issue)):W202–W208, 2009.

[5] P. J. Balwierz, M. Pachkov, P. Arnold, A. J. Gruber, M. Zavolan, and E. van Nimwegen. ISMARA: automated modeling of genomic signals as a democracy of regulatory motifs. *Genome Research*, 24(5):869–884, 2014.

[6] T. Barrett, S. Wilhite, P. Ledoux, and et al. NCBI GEO: archive for functional genomics data sets - update. *Nucleic Acids Research*, 41(Database issue):D991–D995, 2013.

[7] A. Belle, A. Tanay, L. Bitincka, R. Shamir, and E. K. O'Shea. Quantification of protein half-lives in the budding yeast proteome. *Proceedings of the National Academy of Sciences*, 103(35):13004–13009, 2006.

[8] E. Berchtold, G. Csaba, and R. Zimmer. Evaluating transcription factor activity changes by scoring unexplained target genes in expression data. *PLOS ONE*, 11(10), 2016.

[9] E. Berchtold, G. Csaba, and R. Zimmer. RelExplain — integrating data and networks to explain biological processes. *Bioinformatics*, 33(12):1837–1844, 2017.

[10] E. Berchtold, G. Csaba, and R. Zimmer. YESdb: Interactive Integrated Analysis of Stress Datasets. *Database*, submitted.

[11] E. Berchtold, M. Vetter, M. Gündert, G. Csaba, C. Fathke, S. E. Ulbrich, C. Thomssen, R. Zimmer, and E. J. Kantelhardt. Comparison of Six Breast Cancer Classifiers using qPCR. *Bioinformatics*, submitted.

[12] T. Bonfert, E. Kirner, G. Csaba, R. Zimmer, and C. C. Friedel. ContextMap 2: fast and accurate context-based RNA-seq mapping. *BMC Bioinformatics*, 16:122, 2015.

[13] A. Boorsma, B. C. Foat, D. Vis, F. Klis, and H. J. Bussemaker. T-profiler: scoring the activity of predefined groups of genes using gene expression data. *Nucleic Acids Research*, 33((Web Server issue)):W592–W595, 2005.

[14] A.-L. Boulesteix and K. Strimmer. Predicting transcription factor activities from combined analysis of microarray and ChIP data: a partial least squares approach. *Theoretical Biology and Medical Modelling*, 2(1):23, 2005.

[15] J. D. Buenrostro, P. G. Giresi, L. C. Zaba, H. Y. Chang, and W. J. Greenleaf. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nature Methods*, 10:1213–1218, 2013.

[16] H. J. Bussemaker, H. Li, and E. D. Siggia. Regulatory element detection using correlation with expression. *Nature Genetics*, 27(2):167–174, 2001.

[17] S. Cai, C. Luo, J. Thornton, and K. Su. Tailing Local Search for Partial MaxSAT. In *Proc. of AAAI-2014*, pages 2623–2629, 2014.

[18] Cancer Genome Atlas Network and others. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[19] F. Cardoso, L. J. van't Veer, J. Bogaerts, L. Slaets, G. Viale, S. Delaloge, J.-Y. Pierga, E. Brain, S. Causeret, M. DeLorenzi, A. M. Glas, V. Golfinopoulos, T. Goulioti, S. Knox, E. Matos, B. Meulemans, P. A. Neijenhuis, U. Nitz, R. Passalacqua, P. Ravdin, I. T. Rubio, M. Saghatchian, T. J. Smilde, C. Sotiriou, L. Stork, C. Straehle, G. Thomas, A. M. Thompson, J. M. van der Hoeven, P. Vuylsteke, R. Bernards, K. Tryfonidis, E. Rutgers, and M. Piccart. 70-Gene Signature as an Aid to Treatment Decisions in Early-Stage Breast Cancer. *New England Journal of Medicine*, 375(8):717–729, 2016.

[20] Z. Chaim. Conceptual approaches for defining data, information, and knowledge. *Journal of the American Society for Information Science and Technology*, 58(4):479–493, 2007.

[21] A. Chatr-Aryamontri, R. Oughtred, L. Boucher, J. Rust, C. Chang, N. K. Kolas, L. O'Donnell, S. Oster, C. Theesfeld, A. Sellam, C. Stark, B. J. Breitkreutz, K. Dolinski, and M. Tyers. The BioGRID interaction database: 2017 update. *Nucleic Acids Research*, 45(Database issue):D369–D379, Jan 2017.

[22] J. Cox and M. Mann. MaxQuant enables high peptide identification rates, individualized ppb-range mass accuracies and proteome-wide protein quantification. *Nature Biotechnology*, 26(12):1367–1372, 2008.

[23] X. Dai, T. Li, Z. Bai, Y. Yang, X. Liu, J. Zhan, and B. Shi. Breast cancer intrinsic subtype classification, clinical use and future trends. *American Journal of Cancer Research*, 5(10):2929–43, 2015.

[24] J. L. DeRisi, V. R. Iyer, and P. O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278(5338):680–686, 1997.

[25] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[26] ENCODE Project Consortium and others. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[27] F. Erhard and R. Zimmer. Count ratio model reveals bias affecting NGS fold changes. *Nucleic Acids Research*, 43(20):e136, 2015.

[28] J. Ernst, O. Vainas, C. T. Harbison, I. Simon, and Z. Bar-Joseph. Reconstructing dynamic regulatory maps. *Molecular Systems Biology*, 3(1), 2007.

[29] L. Everett, A. Vo, and S. Hannenhalli. PTM-Switchboard–a database of posttranslational modifications of transcription factors, the mediating enzymes and target genes. *Nucleic Acids Research*, 37(Database issue):66–71, Jan 2009.

[30] C. Fan, D. S. Oh, L. Wessels, B. Weigelt, D. S. A. Nuyten, A. B. Nobel, L. J. van't Veer, and C. M. Perou. Concordance among Gene-Expression-Based Predictors for Breast Cancer. *New England Journal of Medicine*, 10(355):560–569, 2006.

[31] D. Fiedler, H. Braberg, M. Mehta, G. Chechik, G. Cagney, P. Mukherjee, A. C. Silva, M. Shales, S. R. Collins, S. van Wageningen, P. Kemmeren, F. C. Holstege, J. S. Weissman, M. C. Keogh, D. Koller, K. M. Shokat, and N. J. Krogan. Functional organization of the S. cerevisiae phosphorylation network. *Cell*, 136(5):952–963, Mar 2009.

[32] E. J. Filardo and P. Thomas. Minireview: G protein-coupled estrogen receptor-1, GPER-1: its mechanism of action and role in female reproductive cancer, renal and vascular physiology. *Endocrinology*, 153(7):2953–2962, 2012.

[33] O. M. Filho, M. Ignatiadis, and C. Sotiriou. Genomic Grade Index: An important tool for assessing breast cancer tumor grade and prognosis. *Critical Reviews in Oncology/Hematology*, 77(1):20–29, 2011.

[34] M. Filipits, M. Rudas, R. Jakesz, P. Dubsky, F. Fitzal, C. F. Singer, O. Dietze, R. Greil, A. Jelen, P. Sevelda, C. Freibauer, V. Müller, F. Jänicke, M. Schmidt, H. Kölbl, A. Rody, M. Kaufmann, W. Schroth, H. Brauch, M. Schwab, P. Fritz, K. E. Weber, I. S. Feder, G. Hennig, R. Kronenwett, M. Gehrmann, and M. Gnant. A new molecular predictor of distant recurrence in ER-positive, HER2-negative breast cancer adds independent information to conventional clinical risk factors. *Clinical Cancer Research*, 17(18):6012–6020, 2011.

[35] K. Fundel, R. Küffner, and R. Zimmer. RelEx–relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371, Feb 2007.

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

[37] A. P. Gasch and M. Werner-Washburne. The genomics of yeast responses to environmental stress and starvation. *Functional & integrative genomics*, 2(4-5):181–192, 2002.

[38] L. Geistlinger, G. Csaba, S. Dirmeier, R. Küffner, and R. Zimmer. A comprehensive gene regulatory network for the diauxic shift in Saccharomyces cerevisiae. *Nucleic Acids Research*, 41(8):8452–8463, 2013.

[39] L. Geistlinger, G. Csaba, R. Küffner, N. Mulder, and R. Zimmer. From sets to graphs: Towards a realistic enrichment analysis of transcriptomic systems. *Bioinformatics*, 27(13):i366–i373, 2011.

[40] D. M. A. Gendoo, N. Ratanasirigulchai, M. S. Schröder, L. Par, J. S. Parker, A. Prat, and B. Haibe-Kains. Genefu: an R/Bioconductor package for computation of gene expression-based signatures in breast cancer. *Bioinformatics*, 32(7):1097, 2016.

[41] Gene Ontology Consortium and others. The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Research*, 32(Database issue):D258–D261, 2004.

[42] S. Ghaemmaghami, W.-K. Huh, K. Bower, R. W. Howson, A. Belle, N. Dephoure, E. K. O'Shea, and J. S. Weissman. Global analysis of protein expression in yeast. *Nature*, 425(6959):737–741, 2003.

[43] A. Goldhirsch, E. P. Winer, A. Coates, R. Gelber, M. Piccart-Gebhart, B. Thürlimann, H.-J. Senn, et al. Personalizing the treatment of women with early breast cancer: Highlights of the st Gallen International Expert Consensus on the Primary Therapy of Early Breast Cancer 2013. *Annals of Oncology*, 24(9):2206–2223, 2013.

[44] C. S. Greene, A. Krishnan, A. K. Wong, E. Ricciotti, R. A. Zelaya, D. S. Himmelstein, R. Zhang, B. M. Hartmann, E. Zaslavsky, S. C. Sealfon, et al. Understanding multicellular function and disease with human tissue-specific networks. *Nature Genetics*, 47(6):569–576, 2015.

[45] J. W. Haefner. *Modeling Biological Systems: Principles and Applications*. Springer Science & Business Media, 2005.

[46] B. Haibe-Kains, C. Desmedt, S. Loi, A. C. Culhane, G. Bontempi, J. Quackenbush, and C. Sotiriou. A three-gene model to robustly identify breast cancer molecular subtypes. *Journal of the National Cancer Institute*, 104(4):311–325, 2012.

[47] B. Haibe-Kains, C. Desmedt, F. Piette, M. Buyse, F. Cardoso, L. van't Veer, M. Piccart, G. Bontempi, and C. Sotiriou. Comparison of prognostic gene expression signatures for breast cancer. *BMC Genomics*, 9(1):394, 2008.

[48] D. Hanahan and R. A. Weinberg. The Hallmarks of Cancer. *Cell*, 100(1):57–70, 2000.

[49] D. Hanahan and R. A. Weinberg. Hallmarks of cancer: The next generation. *Cell*, 144(5):646–674, 2011.

[50] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[51] M. A. Hibbs, D. C. Hess, C. L. Myers, C. Huttenhower, K. Li, and O. G. Troyanskaya. Exploring the functional landscape of gene expression: directed search of large microarray compendia. *Bioinformatics*, 23(20):2692–2699, 2007.

[52] S.-D. Hsu, Y.-T. Tseng, S. Shrestha, Y.-L. Lin, A. Khaleel, C.-H. Chou, C.-F. Chu, H.-Y. Huang, C.-M. Lin, S.-Y. Ho, et al. miRTarBase update 2014: an information resource for experimentally validated miRNA-target interactions. *Nucleic Acids Research*, 42(Database issue):D78–D85, 2014.

[53] S.-S. C. Huang and E. Fraenkel. Integrating proteomic, transcriptional, and interactome data reveals hidden components of signaling and regulatory networks. *Science Signaling*, 2(81):ra40, 2009.

[54] C. A. Hudis, W. E. Barlow, J. P. Costantino, R. J. Gray, K. I. Pritchard, J.-A. W. Chapman, J. A. Sparano, S. Hunsberger, R. A. Enos, R. D. Gelber, and J. A. Zujewski. Proposal for Standardized Definitions for Efficacy End Points in Adjuvant Breast Cancer Trials: The STEEP System. *Journal of Clinical Oncology*, 25(15):2127–2132, 2007.

[55] T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18 Suppl 1:S233–S240, 2002.

[56] R. Jacobson. *Information Design*. Mit Press. MIT Press, 2000.

[57] M. E. Jennex. Re-visiting the knowledge pyramid. In *2009 42nd Hawaii International Conference on System Sciences(HICSS)*, pages 1–7, 2009.

[58] R. M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972.

[59] E. Kawakami, V. K. Singh, K. Matsubara, T. Ishii, Y. Matsuoka, T. Hase, P. Kulkarni, K. Siddiqui, J. Kodilkar, N. Danve, I. Subramanian, M. Katoh, Y. Shimizu-Yoshida, S. Ghosh, A. Jere, and H. Kitano. Network analyses based on comprehensive molecular interaction maps reveal robust control structures in yeast stress response pathways. *System Biology and Applications*, 2(15018), 2016.

[60] P. Khatri and S. Draghici. Ontological analysis of gene expression data: Current tools, limitations, and open problems. *Bioinformatics*, 21(18):3587–95, 2005.

[61] P. Khatri, M. Sirota, and A. J. Butte. Ten years of pathway analysis: Current approaches and outstanding challenges. *PLOS Computational Biology*, 8(2):1–10, Feb 2012.

[62] Y. Kodama, M. Shumway, and R. Leinonen. The sequence read archive: explosive growth of sequencing data. *Nucleic Acids Research*, 40(Database issue):D54–D56, 2012.

[63] A. Kuegel. Improved Exact Solver for the Weighted MAX-SAT Problem. In *POS@ SAT*, pages 15–27, 2010.

[64] R. Küffner, T. Petri, L. Windhager, and R. Zimmer. Petri nets with fuzzy logic (PNFL): reverse engineering and parametrization. *PLOS ONE*, 5(9):e12807, 2010.

[65] D. H. Lackner, M. W. Schmidt, S. Wu, D. a. Wolf, and J. Bähler. Regulation of transcriptome, translation, and proteome in response to environmental stress in fission yeast. *Genome Biology*, 13(4):R25, 2012.

[66] M. V. Lee, S. E. Topper, S. L. Hubler, J. Hose, C. D. Wenger, J. J. Coon, and A. P. Gasch. A dynamic model of proteome changes reveals new roles for transcript alteration in yeast. *Molecular Systems Biology*, 7(1):514, 2011.

[67] J. Leipzig. A review of bioinformatic pipeline frameworks. *Briefings in Bioinformatics*, 18(3):530–536, 2017.

[68] M. D. Leiserson, F. Vandin, H.-T. Wu, J. R. Dobson, J. V. Eldridge, J. L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan, et al. Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes. *Nature Genetics*, 47(2):106–114, 2015.

[69] B. Liu, Y. Han, and S. B. Qian. Cotranslational Response to Proteotoxic Stress by Elongation Pausing of Ribosomes. *Molecular Cell*, 49(3):453–463, 2013.

[70] M. I. Love, W. Huber, and S. Anders. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12):550, Dec 2014.

[71] A. Lundberg, L. S. Lindstrom, J. C. Harrell, C. Falato, J. W. Carlson, P. K. Wright, T. Foukakis, C. M. Perou, K. Czene, J. Bergh, and N. P. Tobin. Gene expression signatures and immunohistochemical subtypes add prognostic value to each other in breast cancer cohorts. *Clinical Cancer Research*, 23(24):7512–7520, 2017.

[72] K. Mitra, A.-R. Carvunis, S. K. Ramesh, and T. Ideker. Integrative approaches for finding modular structure in biological networks. *Nature Reviews Genetics*, 14(10):719–732, 2013.

[73] C. Mitrea, Z. Taghavi, B. Bokanizad, S. Hanoudi, R. Tagett, M. Donato, C. Voichia, and S. Draghici. Methods and approaches in the topology-based analysis of biological pathways. *Frontiers in Physiology*, 4:278, 2013.

[74] F. Miura, N. Kawaguchi, M. Yoshida, C. Uematsu, K. Kito, Y. Sakaki, and T. Ito. Absolute quantification of the budding yeast transcriptome by means of competitive PCR between genomic and complementary DNAs. *BMC Genomics*, 9:574, 2008.

[75] K. A. Morano, C. M. Grant, and W. S. Moye-Rowley. The response to heat shock and oxidative stress in Saccharomyces cerevisiae. *Genetics*, 190(4):1157–1195, 2012.

[76] S. Paik, S. Shak, G. Tang, C. Kim, J. Baker, M. Cronin, F. L. Baehner, M. G. Walker, D. Watson, T. Park, W. Hiller, E. R. Fisher, D. L. Wickerham, J. Bryant, and N. Wolmark. A multigene assay to predict recurrence of tamoxifen-treated, node-negative breast cancer. *The New England Journal of Medicine*, 351(27):2817–26, 2004.

[77] J. S. Parker, M. Mullins, M. C. Cheang, S. Leung, D. Voduc, T. Vickery, S. Davies, C. Fauron, X. He, Z. Hu, J. F. Quackenbush, I. J. Stijleman, J. Palazzo, J. Marron, A. B. Nobel, E. Mardis, T. O. Nielsen, M. J. Ellis, C. M. Perou, and P. S. Bernard. Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes. *Journal of Clinical Oncology*, 27(8):1160–1167, 2009.

[78] A. Prat, J. S. Parker, C. Fan, M. C. U. Cheang, L. D. Miller, S. K. L. Bergh, J a nd Chia, P. S. Bernard, T. O. Nielsen, M. J. Ellis, L. A. Carey, and C. M. Perou. Concordance among gene expression-based predictors for ER-positive breast cancer tr eated with adjuvant tamoxifen. *Annals of Oncology*, 23(11):2866–2873, 2012.

[79] J. Ptacek and M. Snyder. Charging it up: global analysis of protein phosphorylation. *Trends in Genetics*, 22(10):545–554, Oct 2006.

[80] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

[81] E. Rintala, P. Jouhten, M. Toivari, M. G. Wiebe, H. Maaheimo, M. Penttilä, and L. Ruohonen. Transcriptional responses of Saccharomyces cerevisiae to shift from respiratory and respirofermentative to fully fermentative metabolism. *OMICS*, 15(7-8):461–476, 2011.

[82] J. Rowley. The wisdom hierarchy: representations of the DIKW hierarchy. *Journal of Information Science*, 33(2):163–180, 2007.

[83] A. Sandelin, W. Alkema, P. Engström, W. W. Wasserman, and B. Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, 32(Database issue):D91–D94, 2004.

[84] P. Shah, Y. Ding, M. Niemczyk, G. Kudla, and J. B. Plotkin. Rate-limiting steps in yeast protein translation. *Cell*, 153(7):1589–1601, 2013.

[85] R. Shalgi, J. A. Hurt, I. Krykbaeva, M. Taipale, S. Lindquist, and C. B. Burge. Widespread Regulation of Translation by Elongation Pausing in Heat Shock. *Molecular Cell*, 49(3):439–452, 2013.

[86] N. Shedroff. Information interaction design: A unified field theory of design. *Information design*, pages 267–292, 1999.

[87] L. H. Sobin, M. K. Gospodarowicz, and C. Wittekind. *TNM classification of malignant tumours*. John Wiley & Sons, 2011.

[88] L. Song and G. Crawford. DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harb Protoc*, 2, Feb 2010.

[89] J. A. Sparano, R. J. Gray, D. F. Makower, K. I. Pritchard, K. S. Albain, D. F. Hayes, C. E. Geyer, E. C. Dees, E. A. Perez, J. A. Olson, J. Zujewski, T. Lively, S. S. Badve, T. J. Saphner, L. I. Wagner, T. J. Whelan, M. J. Ellis, S. Paik, W. C. Wood, P. Ravdin, M. M. Keane, H. L. Gomez Moreno, P. S. Reddy, T. F. Goggins, I. A. Mayer, A. M. Brufsky, D. L. Toppmeyer, V. G. Kaklamani, J. N. Atkins, J. L. Berenberg, and G. W. Sledge. Prospective Validation of a 21-Gene Expression Assay in Breast Cancer. *New England Journal of Medicine*, 373(21):2005–2014, 2015.

[90] S. L. Spurgeon, R. C. Jones, and R. Ramakrishnan. High Throughput Gene Expression Measurement with Real Time PCR in a Microfluidic Dynamic Array. *PLOS ONE*, 3(2):1–7, Feb 2008.

[91] C. Stoll. *Silicon Snake Oil: Second Thoughts on the Information Highway*. Anchor books. Anchor Books, 1996.

[92] A. Subramanian, H. Kuehn, J. Gould, P. Tamayo, and J. P. Mesirov. GSEA-P: a desktop application for Gene Set Enrichment Analysis. *Bioinformatics*, 23(23):3251–3253, 2007.

[93] M. C. Teixeira, P. T. Monteiro, J. F. Guerreiro, J. P. Gonçalves, N. P. Mira, S. C. dos Santos, T. R. Cabrito, M. Palma, C. Costa, A. P. Francisco, et al. The YEASTRACT database: an upgraded information system for the analysis of gene and genomic transcription regulation in Saccharomyces cerevisiae. *Nucleic Acids Research*, 42(Database issue):D161–D166, 2013.

[94] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10):6567–6572, 2002.

[95] M. Tsompana and M. J. Buck. Chromatin accessibility: a window into the genome. *Epigenetics & Chromatin*, 7(1):33, Nov 2014.

[96] D. Venet, J. E. Dumont, and V. Detours. Most random gene expression signatures are significantly associated with breast cancer outcome. *PLOS Computational Biology*, 7(10), 2011.

[97] J. Verghese, J. Abrams, Y. Wang, and K. a. Morano. Biology of the heat shock response and protein chaperones: budding yeast (Saccharomyces cerevisiae) as a model system. *Microbiology and Molecular Biology Reviews*, 76(2):115–58, 2012.

[98] J. A. Vizcaíno, A. Csordas, N. Del-Toro, J. A. Dianes, J. Griss, I. Lavidas, G. Mayer, Y. Perez-Riverol, F. Reisinger, T. Ternent, Q. W. Xu, R. Wang, and H. Hermjakob. 2016 update of the PRIDE database and its related tools. *Nucleic Acids Research*, 44(Database issue):D447–D456, 2016.

[99] G. Von Minckwitz, M. Untch, J. U. Blohmer, S. D. Costa, H. Eidtmann, P. A. Fasching, B. Gerber, W. Eiermann, J. Hilfrich, J. Huober, C. Jackisch, M. Kaufmann, G. E. Konecny, C. Denkert, V. Nekljudova, K. Mehta, and S. Loibl. Definition and impact of pathologic complete response on prognosis after neoadjuvant chemotherapy in various intrinsic breast cancer subtypes. *Journal of Clinical Oncology*, 30(15):1796–1804, 2012.

[100] K. Wanichthanarak, I. Nookaew, and D. Petranovic. yStreX: yeast stress expression database. *Database*, 2014, 2014.

Acknowledgements

There are many people that have supported and encouraged me in various ways while I was working on this thesis.

First of all, I want to thank Prof. Ralf Zimmer for giving me the opportunity to work on so many interesting topics and for the helpful discussions that always improved my research.

I would like to thank Prof. Jan Baumbach for reviewing this thesis and Prof. Christian Böhm for being chairman of my dissertation committee; our collaboration partner of the Martin-Luther University in Halle and the Buchner lab for providing the measurements for the breast cancer subtype classification and yeast heat shock projects.

I am very grateful to Gergely Csaba who always had an open ear for any problems I encountered and in most cases was able to suggest a solution and to Tobias Petri who supervised my work as a student assistant and encouraged me to challenge myself with interesting bioinformatic problems. I also thank all other colleagues for creating an inspiring work atmosphere.

Finally, my deepest thanks are to my family and especially my grandparents for supporting me and always having an interest in my work, even though I could not really explain to them what I was doing. Last, I want to thank Thomas for his love, his enthusiasm and for always knowing what I need.

Lebenslauf

Persönliche Daten

Name	Evi Berchtold
Geburtsdatum	10.03.1988
Geburtsort	München

Ausbildung

1996-2007	Abitur, Feodor-Lynen Gymnasium Planegg
2007-2010	Bachelor Bioinformatik Ludwig-Maximilians-Universität München Technische Universität München
2010-2012	Master Bioinformatik Ludwig-Maximilians-Universität München Technische Universität München

Stipendium

2007-2012	e-fellows Stipendium

Beruflicher Werdegang

2009-2012	Studentische Hilfskraft LFE Bioinformatik, Ludwig-Maximilians-Universität München
seit 2012	wissenschaftliche Mitarbeiterin LFE Bioinformatik, Ludwig-Maximilians-Universität München

Publikationen

Sebastian Dintner, Anna Staron, **Evi Berchtold**, Tobias Petri, Thorsten Mascher, Susanne Gebhard (2011). Coevolution of ABC transporters and two-component regulatory systems as resistance modules against antimicrobial peptides in Firmicutes bacteria. Journal of Bacteriology, 193(15), 3851-3862.

Tobias Petri*, **Evi Berchtold***, Ralf Zimmer, Caroline C. Friedel (2012). Detection and correction of probe-level artefacts on microarrays. BMC Bioinformatics, 13(1), 114.

Evi Berchtold, Gergely Csaba, Ralf Zimmer (2016). Evaluating Transcription Factor Activity Changes by Scoring Unexplained Target Genes in Expression Data. PloS ONE, 11(10), e0164513.

Evi Berchtold, Gergely Csaba, Ralf Zimmer (2017). RelExplain - integrating data and networks to explain biological processes. Bioinformatics, 33(12), 1837-1844.

Constantin Ammar, **Evi Berchtold**, Gergely Csaba, Andreas Schmidt, Axel Imhof, Ralf Zimmer (2017). Multi-reference spectral library yields almost complete coverage of heterogeneous LC-MS/MS data sets. bioRxiv, 180448.

Manuskripte

Evi Berchtold, Gergely Csaba, Ralf Zimmer. YESdb: Interactive Integrated Analysis of Stress Datasets. Database, *Submitted*

Evi Berchtold, Martina Vetter, Melanie Maierthaler, Gergely Csaba, Christine Fathke, Susanne Ulbrich, Christoph Thomssen, Eva Johanna Kantelhardt, Ralf Zimmer. Comparison of Six Breast Cancer Classifiers using qPCR. Bioinformatics, *Submitted*

Christoph Stratil*, **Evi Berchtold***, Gergely Csaba, Moritz Mühlhofer, Nina Bach, Stephan Sieber, Martin Haslbeck, Ralf Zimmer, Johannes Buchner. The heat shock response is a two-pronged system.

---

*Gemeinsame Erstautoren