# Gene and Protein Networks in Understanding Cellular Function

Sonja Katriina Lehtinen

A thesis sumbitted to
University College London
for the degree of
Doctor of Philosophy

March 2015

I, Sonja Lehtinen, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Sonja Lehtinen
2 March, 2015

# Abstract

Over the past decades, networks have emerged as a useful way of representing complex large-scale systems in a variety of fields. In cellular and molecular biology, gene and protein networks have attracted considerable interest as tools for making sense of increasingly large volumes of data. Despite this interest, there is still substantial debate over how to best exploit network models in cellular biology. This thesis explores the use of gene and protein networks in various biological contexts.

The first part of the thesis (Chapter 2) examines protein function prediction using network-based 'guilt-by-association' approaches. Given the falling costs of genome sequencing and the availability of large volumes of biological data, automated annotation of gene and protein function is becoming increasingly useful. Chapter 2 describes the development of a new network-based protein function prediction method and compares it to a leading algorithm on a number of benchmarks. Biases in benchmarking methods are also explicitly explored.

The second part (Chapters 3 and 4) explores network approaches in understanding loss of function variation in the human genome. For a number of genes, homozygous loss of function appears to have no detrimental effect. A possible explanation is that these genes are only necessary in specific genetic backgrounds. Chapter 3 develops methods for identifying these types of relationships between apparently loss of function tolerant genes. Chapter 4 describes the use of networks in predicting the functional effects of loss of function mutations.

The third part of the thesis (Chapters 5 and 6) uses network representations to model the effects of cellular stress on yeast cells. Chapter 5 examines stress induced changes in co-expression and protein interaction networks, finding evidence of increased modularisation in both types of network. Chapter 6 explores the effect of stress on resilience to node removal in the co-expression networks.

# Acknowledgements

If theses are written from the shoulders of giants, I have been lucky to work with particularly big and friendly ones. First and foremost, I would like to thank my primary supervisor Christine Orengo for her outstanding guidance and genuine kindness. I am also grateful for the excellent input and good humour of my second supervisor Jürg Bähler and the support of my thesis chair Konstantinos Thalassinos.

I have had the pleasure of collaborating with a number of excellent scientists during this PhD. My thesis benefited greatly from the expertise and insight of John Shawe-Taylor, Jon Lees, Vera Pancaldi, Suganthi Balasubramanian and Koon-Kiu Yan. I would also like to thank everybody in the Orengo group for their help along the way.

In my final year, I had the chance to visit the Gerstein Lab at Yale for three months. This was a fantastic experience - a big thank you to Mark Gerstein, UCL graduate school and the Yale-UCL Collaborative for making my visit possible and all of the Gerstein Lab for their friendly welcome.

This PhD would not have been possible without the financial and academic support of CoMPLEX or the friendship of the CoMPLEX people. I could not have asked for a better PhD cohort.

Finally, to my family and the rest of my friends: thank you for rarely asking about the PhD - it was appreciated.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Overview: Network Biology

A *network* or *graph* is a mathematical representation of a set of entities (*nodes*) and the relationships (*edges*) between them. From a mathematical point of view, networks have been of interest for a long time: early proofs in graph theory date as far back as the 1700s. The use of network representations in the sciences also has a rich history: they have long been used in a variety of fields to model diverse structures, ranging from social systems to atomic interactions.

In the past decades however, the study of networks has undergone significant changes. Increased computational resources have allowed us to shift our focus from small-scale networks and the properties of individual nodes to the study of complex large-scale networks. Interest in these larger networks has driven development of *complex network theory*, a field aiming to characterise, model and predict the structure, properties and behaviour of these network systems [159]. The applicability of this approach is not restricted to a single field of study - large and complex networks are equally relevant in physics as they are in social sciences. This multidisciplinarity has led to hopes that universal laws governing the behaviour of complex networks will emerge [14].

Network approaches have been popular in biology, particularly at the level of gene or protein networks. At least two factors have contributed to this surge of interest. Firstly, over the last two decades, there have been marked advances in high-throughput experimental technologies (*'omics' methodologies*) and the computational resources to store and manipulate large data sets. This has led to an unprecedented wealth of biological data. Networks often provide a convenient and efficient way of conceptualising these large data sets. Furthermore, more detailed representations, such as systems of dynamical equations for example, become impractical for very large systems, leading many authors to favour the simpler network models [69]. Secondly, the past few decades have also seen the emergence of *systems biology* - research approaches seeking to understand biological function in terms of the interacting components of biological systems. Network representations are well suited to this research approach.

The specific methodologies applied to the study of gene and protein networks have been numerous and varied. Fundamentally, however, these diverse approaches share the same central idea: there is a connection between the topology and function of gene and protein networks - the study of topology can therefore help us understand function. Traditionally, graph theorists have focused on networks with either completely regular (where each node has the same number of neighbours) or completely random (where the probability of any two nodes being connected is constant across the network) connectivities. The structure of gene and protein networks appears to lie somewhere in between these extremes (Figure 1.1) [246]. This opens up two interesting avenues of research: understanding the function of a specific node in relation to its position in the network and understanding the function of the network as a whole in light of its topology.

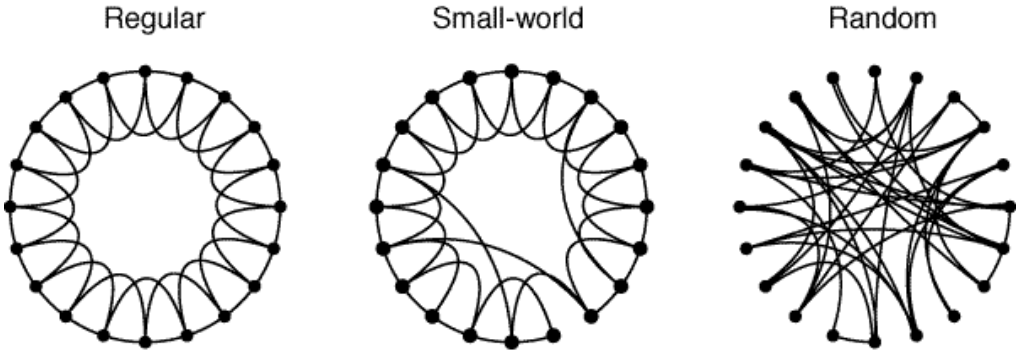


Figure 1.1: Illustration of regular lattice, random and so called 'small-world' networks. Small world networks are generated by randomly moving ('re-wiring') a proportion of the lattice network's edges. Small-world networks display some key properties of real-world networks such as a highly clustered structure, combined with relatively small average shortest path lengths. Image from Watts et al [246].

## 1.2 Network Concepts and Terminology

A *network*, sometimes referred to as a graph, is a mathematical object describing the relationships between a set of entities. The entities are referred to as *nodes*, while the relationships between them are *edges*. Complex networks are graphs with non-trivial topological features. Some networks consist of multiple unconnected sub-networks: these sub-networks are referred to as network *components*.

Edges can describe relationships with or without directionality - networks are referred to as *directed* or *undirected* accordingly. For example, transcriptional regulatory networks are directed: there is a distinction between regulating and being regulated by. Protein binding networks, on the other hand, are undirected, because binding relationships are symmetrical. A *weighted* network is one in

which edges are associated with a numerical value $w$, describing some property of the edge, such as the strength of an interaction for example.

Networks are often represented in the form of an *adjacency matrix*, $A$, where:

$$A(i,j) = \begin{cases} w & \text{if } i \text{ and } j \text{ are joined by an edge with weight } w, \\ 0 & \text{if } i \text{ and } j \text{ are not joined by an edge.} \end{cases}$$

A number of measures have been developed to describe properties of entire networks as well as properties of individual nodes and edges within a network. The paragraphs below briefly summarise the most commonly used properties.

The *degree* $k$ of a node is the number of other nodes it is connected to. The *degree distribution*, $P(k)$ gives the probability that a randomly selected node in the network has degree $k$. For directed networks, authors often differentiate between *out-degree* (connections originating from the node) and *in-degree* (connections from other nodes to the node), with corresponding in- and out-degree distributions. For weighted networks, the *weighted degree* of a node refers to the sum of its edges' weights.

The *shortest path length* or *geodesic* is the minimum number of steps needed to move from one node to another in the network. The *average shortest path length* is the mean shortest path length between all node pairs in the graph and thus gives an indication of global connectivity. Network *diameter* is the length of the single longest geodesic in the network.

A drawback of using path lengths is that the measure does not cope with disconnected graphs particularly well: the path length between nodes in different components is infinite, rendering the average measure meaningless. Thus, some authors prefer to use *efficiency*, the reciprocal of the geodesic and, correspondingly *global efficiency*, the average of the reciprocals of all shortest path lengths in the network. Despite this advantage, this measure is still relatively rare within the field, perhaps because it is less intuitive than shortest path length. Calls have been made for the increased use of efficiency rather than geodesic [159].

The centrality of nodes in the network is often of interest and there are a number of ways of measuring this property. These include *betweenness centrality*, the number of shortest paths in the network running through the node; *closeness centrality*, the reciprocal of the average shortest path lengths from the node to all others in the network and *eigenvector centrality*, computed, for the $i^{th}$ node as the $i^{th}$ component of the principal eigenvector of the adjacency matrix.

Other measures used to describe the properties of the network as a whole include the *clustering coefficient* or *transitivity*, the probability that a node's neighbours are also connected and *assortativity*, the correlation between the properties (typically degree) of connected nodes.

A network *module* or *community*, in general terms, indicates a group of nodes that have a higher density of connections to each other than to the rest of the

network. Examples of modules cover, for example, friendship groups in social networks or protein complexes in protein networks. While highly intuitive, the concept of a network module lacks a precise definition. There are a variety of module finding algorithms, each using a different specification for what type of module is being sought.

## 1.3 Network Approaches

This section will review how complex network theory is applied in the study of biological networks. We will first discuss how the measures outlined above are used to characterise nodes and networks and then focus on the development of network models and how these have been used to gain functional insight from gene and protein networks.

### 1.3.1 Characterising Nodes

Early work on networks was concerned with characterising the properties of individual nodes - for example, by identifying key players in large social networks. While node-focused approaches have become impractical for very large networks [159], they remain relevant for gene and protein networks.

A large part of node-centric approaches have sought to relate the position of a node in a network to its functional importance, such as, for example, a gene's essentiality. The earliest work in this field used protein interaction networks to predict the lethality of mutations in yeast genes: genes with high network centrality were found to be more likely to be essential for survival [102]. Since then, similar approaches have been applied to different types of network [167], in different organisms [75] and using various types of centrality measures [172, 254].

To some extent, the relationship between essentiality and lethality may not be as straightforward as first thought: some authors have reported negative results [253] while others have questioned which measures best capture the relationship [172]. Furthermore, the effect might be partially an artefact due to sampling biases in interactome mapping. High-throughput protein interaction detection techniques have been found to favour highly expressed and highly conserved proteins [239], both of which are also likely to be features of essential proteins. Furthermore, if data from small-scale studies is also included, the well studied genes are more likely to have a higher number of connections. Because essential genes are more likely to be well studied, the connection between essentiality and centrality may therefore be at least partly due to biases in the data. Despite these concerns, a recent comprehensive study suggests that the relationship between lethality and centrality holds for both degree and betweenness centrality in a wide range of organisms [189].

More recent work has sought to relate the characteristics of genes in a network to functional properties beyond essentiality. For example, centrality measures

have been used to predict disease related genes [164] and to study the adverse effect of drugs: the degree and centrality of a drug's non-intended targets are predictive of the number of side effects it has [242]. Furthermore, new measures of node properties have been introduced in an attempt to capture characteristics relating to other aspects of function. For example, Hwang et al. used *bridging centrality*, the extent to which a node acts as a connector between two network modules, to identify potential modulators of information flow between different biological processes [93].

Another example of research strategies involving the study of individual nodes within the context of the network are *guilt-by-association* approaches to protein function prediction. The rationale behind these methods is that binding, co-expression, co-localisation and other relationships between genes and proteins can be considered evidence of functional association. Therefore, networks can be used to infer what functionally uncharacterised proteins do, or to suggest new players in established pathways. Early prediction algorithms focused on direct network neighbourhood, but more sophisticated strategies, taking into account the wider network topology, have been developed since then. These approaches have also been successfully applied in clinical settings: network-based biomarkers for disease diagnosis have also been developed, for example in breast cancer metastasis [33].

### 1.3.2 Characterising Networks

A second approach to the study of networks is attempting to link the topology of the network as a whole to the function of the cell, instead of focusing on individual genes or proteins.

In general terms, real-world complex networks, including gene and protein networks, share a number of characteristics differentiating them from 'random' networks: real-world networks, compared to random networks, tend to have short geodesics ('small-world' property), heavy-tailed degree distributions, high clustering coefficient, high assortitivity and a highly modular structure [159].

The study and interpretation of the heavy-tailed degree distributions in particular has attracted a significant amount of attention: there has been considerable debate over the role and meaning of this property. In early literature on biological network topology, heavy-tailed degree distributions were often reported as 'power law' or 'scale-free' distributions: the probability of a node having degree $k$ was reported to follow $P(k) = ak^{\gamma}$, where $\gamma$ and $a$ are constants (Figure 1.2). However, these claims were often based on visual inspection and lacked statistical support [137] - indeed, when appropriate goodness of fit measures were applied on a sample of ten networks reported as 'scale-free' in the literature, none of the claims were found to be statistically robust [110].

In a number of contexts, it may not be particularly important whether the distribution fits a power law. The presence of a heavy-tail (power law distributed

or not) implies the presence of nodes with very high connectivity, which is functionally interesting in itself. However, the emergence of power laws has been considered particularly interesting because, in statistical physics, power law behaviour observed in macroscopic phenomena arises from laws operating at the microscopic scale [220]. This has led to speculation that similar laws could be identified in biological systems as well. Thus, enthusiasm for power laws may have been partially driven by a desire to 1) find universal properties that transcend the specific system under study [249] and 2) in the context of biological systems, find unifying laws or generative mechanisms that explain how these laws arise [14].

In the context of gene and protein networks, is not always clear what the biological implications of the observations about distribution are. Indeed, simply identifying power laws, even when statistically sound, does not necessarily imply an interesting generative mechanism is at work: by an extension of the central limit theorem, the sum of multiple variables drawn from heavy-tailed, but not necessarily power law distributions, is power law distributed [249]. Thus, even where power law distributions are correctly observed, they may not be indicative of underlying unifying laws, but simply arise as a by-product of mixing multiple distributions [220]. Considering the generative mechanism behind the observed distribution is therefore crucial.

Despite these concerns, there have been interesting results in this field, particularly in the context of network growth and evolution. Barabasi and Albert proposed the *preferential attachment* [13] model of network growth to explain the degree distributions observed in real-world networks. The model is based on the idea that the probability of a new node attaching to node $i$ is proportional to the degree of $i$. A similar idea is neatly applicable to protein networks, if we assume they grow by gene duplication and divergence [96]: new genes arise as modified copies of existing genes, which inherit the original gene's interactions with some probability. Thus, the more interactions a gene has, the likelier it is to develop new ones, because the probability that one of its partners will be duplicated is high. While Barabasi and Albert developed the model in the context of power law distributions, the principle, if not the detail of their model, is applicable to heavy-tailed distributions more generally.

Another area where overall network topology has promised functional insight is the study of *network robustness*, the network's ability to maintain normal function in face of perturbation. Various authors have suggested that the topology of the network plays an important role in determining its robustness to node removal (a model for loss of function mutation in gene and protein networks): networks with power law distributions tolerate removal of a higher proportion of their nodes before disintegrating than random networks [3]. Other authors have suggested that the modularity of biological networks is also a robustness maximising strategy: relatively independent functional modules would minimise

the spreading of the perturbation to the network as a whole [115]. Overall, these suggestions imply that robustness has been an important factor in the evolution of networks - indeed simulation of possible *Escherichia coli* (*E. coli*) chemotaxis signalling network topologies suggest that the true network is the smallest sufficiently robust network [118].

### 1.3.3 Modelling Networks

The mathematical modelling of networks and network processes is a growing research area [159]. The aim is to construct statistical models of networks that capture the character of real-world networked systems. The development of representative statistical models would aid the development of a principled framework for studying empirical networks. Specifically, it has been suggested they could guide the development of meaningful network metrics, help us understand how these metrics relate to the behaviour of the network and allow prediction of this behaviour [159]. In the context of biological networks, accurate statistical models of network structure could also provide insights into how the network has evolved [184], help optimise the discovery of new interactions by guiding the choice of proteins to study [128] and allow the generation of synthetic datasets for testing and perfecting computational algorithms [83].

Here, we will briefly discuss some of the main network models that have been employed in the study of gene and protein networks.

Perhaps the first attempt at constructing a model of a large-scale network was the 'random network', introduced in the context of social networks by Solomonoff and Rapoport [215] and, later (independently) by Erdös and Rényi [50]. The *Erdös-Rényi* (ER) random graph, as this model is often referred to, is constructed by taking a set of $n$ nodes and connecting each pair with probability $p$. This results in a network, where, in the limit of large $n$, the probability of a node having degree $k$ follows a Poisson distribution:

$$P(k) = \binom{n-1}{k} p^k (1-p)^{(n-1-k)} \simeq \frac{z^k e^{-z}}{k!}$$

where $z$ is the mean degree $p(n-1)$.

The ER network has been extensively studied and many of its properties are well characterised. While well understood, the ER network is an inadequate model of real-world networks: it fails to capture many of the key properties of real-world networks. A particularly significant shortfall of the model is the degree distribution: the Poisson degree distribution lacks the heavy tail of real-world degree distributions [159] (Figure 1.2). Other differences include lack of clustering, assortativity and community structure in ER networks [159]. Thus, in the context of gene and protein networks, ER models are mainly used to contrast with more realistic network models (see, for example [3]).

The *configuration model* allows network models with more realistic degree

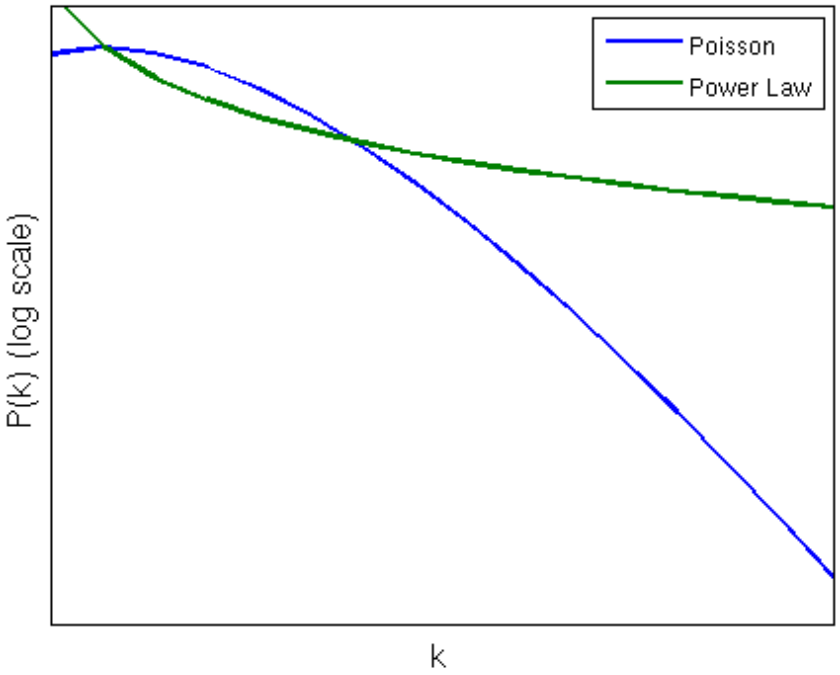


Figure 1.2: Illustration of degree distributions ($P(k)$) for 'random' networks (Poisson degree distribution $P(k) = \frac{z^k e^{-z}}{k!}$, where $z$ is the mean degree) and 'scale-free' (power law degree distribution $P(k) = ak^{\gamma}$, where $a$ and $\gamma$ are constants) with the same average degree. Random networks are generally used to contrast with networks with more realistic degree distributions. Gene and protein networks generally have heavy-tailed degree distributions (though not necessarily following a power law). To some extent, this property may be due to biases in network detection algorithms.

distributions [159]. The network is generated by defining a degree sequence (the sequence of $n$ values of degrees $k_i$ for nodes $i = 1, ..., n$). We can think of this as giving each node $k_i$ 'stubs' and then randomly drawing edges between the stubs, achieving a network with the predefined degree sequence. In practice, configuration models are often used as null models for empirical networks: we are often interested in how properties of an observed network differ from a random network with the same degree configuration. This approach has, for example, been applied to assessing network modularity in the context of network clustering [92].

More sophisticated generalisations of these models exist (see, for example, [18]) - a common problem, however, is that none of these methods capture the high clustering coefficient often observed in real-world networks.

Some models have specifically attempted to capture this property, for example, *small-world models* [246]. These models are based on starting with a regular lattice graph and randomly rewiring a proportion of its edges. Depending on the proportion of edges rewired, the resulting network will fall somewhere between a

regular lattice structure and a random network. These networks have generated a lot of interest among theoreticians [158], but are rarer in the biological literature, perhaps because the generative mechanism (rewiring edges in a lattice graph) does not seem realistic in a biological context.

Interestingly, a somewhat related class of models, *geometric random graphs* have been proposed in biological contexts. In these models, nodes are placed randomly in space - for example, in the two dimensional case, nodes are randomly assigned $x$ and $y$ coordinates drawn independently from the uniform (0,1) distribution. Each pair of nodes is then connected if the distance (typically Euclidean distance) between them is smaller than some parameter value. These networks capture many of the properties of real-world protein-protein interaction networks, including measures of connectivity and clustering [83,183]. Pržulj et al. have proposed a biological interpretation of these models: the space in which the proteins are embedded represents their biochemical properties. This interpretation allows modelling network growth in terms of gene duplication and mutation: the duplicated gene starts at the same location as the 'parent' gene and then acquires mutations and moves away from the parent, thus inheriting some of its parent's interactions [184]. This model relies on the assumption that interactions occur between proteins with similar biochemical properties - it is unclear whether there is any evidence to support this idea. For example, a trivial prediction of the model is that protein bind themselves - which is not the case for a majority of proteins.

#### Future directions

Despite progress in the field, there are still a number of open research questions [159]. There is as yet no clear consensus on which network characteristics best capture functionally relevant information about gene and protein networks and the extent to which this depends on the network or aspect of function being studied. A related open question is the extent to which observed properties of gene and proteins networks reflect genuine biology, as opposed to resulting from biases in the way these networks are generated. Finally, none of the network models proposed so far adequately capture the properties of gene and protein networks while also having a plausible biological interpretation.

## 1.4 Network Types

In gene and protein networks, the nature of the nodes is clear: they represent either genes or gene products. The relationship captured by the edges usually reflects some form of functional association between the nodes. This section summarises how the most well studied gene and protein networks are mapped and analysed. The major repositories holding various types of network data are summarised in Table 1.1.

| Name | Interactions | Organisms | Notes |
|---|---|---|---|
| BioGRID | Physical (experimental); Genetic | Numerous (eukaryotic, prokaryotic and viral) | |
| IntAct | Physical (experimental) | Numerous (eukaryotic, prokaryotic) | |
| MINT | Physical (experimental) | Numerous (eukaryotic, prokaryotic and viral) | Various related databases, such as HomoMINT, a human interaction network with homology-based predicted interactions. |
| DIP | Physical (experimental) | Numerous | |
| I2D | Physical (experimental) | Human, fly, mouse, rat, worm, yeast | Integrates information across various other databases. |
| iRefIndex | Physical (experimental); Genetic | Numerous (eukaryotic, prokaryotic) | Integrates information across various other databases. |
| STRING | Physical (experimental); Predicted (various methods) | Numerous (eukaryotic, prokaryotic) | Interactions are weighted according to estimated reliability. |
| PIPs | Physical (predicted) | Human | Interactions are weighted according to estimated reliability. |
| KEGG | Signalling pathway; Metabolic pathway | Numerous (eukaryotic, prokaryotic) | Also contains non-interaction data – including information relating to drugs, disease and ontology groups. |

Table 1.1: Summary of publicly available repositories for various types of network data.

### 1.4.1 Protein-Protein Interaction Networks

Protein-protein interaction (PPI) networks depict physical binding between proteins and are among the most available and well studied molecular interaction networks [99]. The specific form of the interaction captured depends on the data-source: protein binding may be stable or transient and interactions may depict binary association between proteins or alternatively represent protein complex co-membership. Although protein-protein interactions are conceptually straightforward, their detection can be difficult and different experimental techniques may introduce different forms of bias. It is therefore important to have an understanding of the techniques used to map protein-protein interactions.

#### Experimental Techniques

There are a number of different experimental techniques for identifying protein-protein interactions. In broad terms, approaches fall into one of two categories: genetic and biochemical approaches [59].

Genetic approaches are based on modifying the proteins of interest so that their interaction produces a detectable signal. Genetic techniques are therefore suited to mapping binary interactions and are generally capable of detecting transient, as well as stable, binding.

Yeast two hybrid screening [98] is among the widest used genetic detection techniques. The two genes of interest, often referred to as *bait* and *prey*, are modified to include the activation and binding domains of a transcription factor. As illustrated in Figure 1.3, if the proteins interact, the activation and binding domains are brought into close proximity, producing a functional transcription factor, which will lead to transcription of a reporter gene. This allows the interaction to be detected. The disadvantage of this approach is that interactions will only be found if they occur in the nucleus [38] and screens are vulnerable to other sources of noise, such as mis-folding of the transcription factor [185]. Other examples of genetic techniques include LUMIER [15], a similar technique developed for mammalian cells, where baits are tagged with a luciferase and prey with a FLAG tag (protein sequence recognised by an antibody) so that interactions can be detected by a luciferase assay on anti-Flag immunoprecipitates; and fragment complementation assays (PCA), in which the genes of interest are fused with complementary fragments of a reporter protein [224].

Biochemical methods [61, 86], such as tandem affinity purification followed by mass-spectrometric protein complex identification (TAP-MS) [198], provide a complementary approach to interaction mapping: these methods focus on identifying protein complexes. Although variations on the technique exist, the general principle is that a protein of interest is fused with a TAP tag, allowing the protein and its binding partners to be purified through affinity selection (Figure 1.4). Binding partners can then be identified through mass-spectrometry. The disadvantage of these methods is that they are vulnerable to tagging disrupting

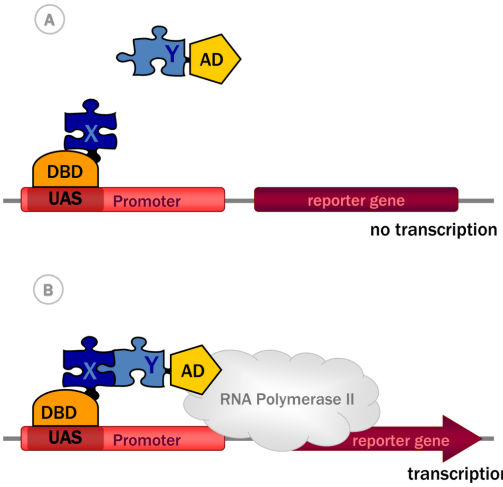


Figure 1.3: Illustration of a yeast two hybrid system. The bait (X), is fused to the DNA binding domain. A potential interactor or prey (Y) is fused to the activation domain (AD) The interaction of the bait and prey leads to reconstruction of a functional transcription factor, recruitment of RNA polymerase and transcription of the reporter gene. Figure reproduced from [24].

complex formation and to weakly associated components dissociating from the complex during the purification process [185].

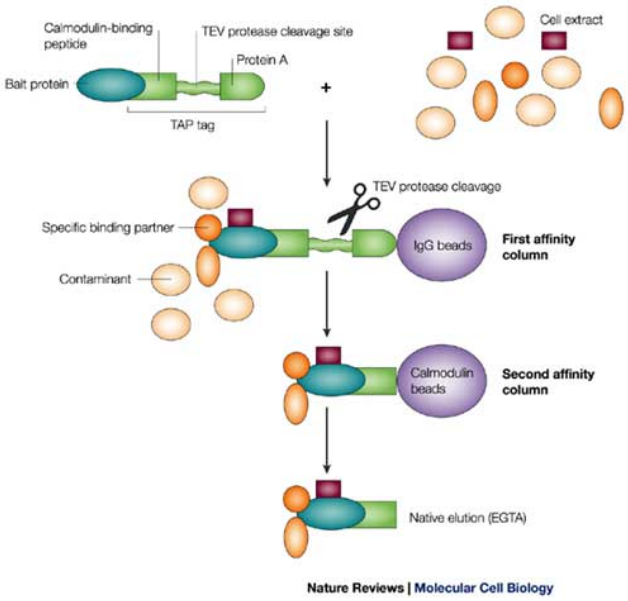


Figure 1.4: Illustration of tandem affinity purification. The protein of interest (bait) is fused with a TAP tag, allowing it and its binding partners to be isolated through affinity purification. Figure reproduced from [91]

### Data Quality

Assessing the quality of high throughput protein-protein interactions is a key step in using these data to make biological inferences. Estimating PPI data quality, however, is not necessarily trivial. A number of factors contribute to

the reliability of an experimental data set: the detection technique's precision (the fraction of detected interactions that are true positives), its sensitivity (the fraction of true positives that the technique is able to detect) and any systematic biases in favour or against particular types of interaction.

One approach for determining false negative and false positive rates is to examine the extent of overlap between different interaction data sets. Von Mering et al. found that, out of 80000 yeast protein interactions identified by various high throughput techniques, only 2400 were identified by more than one method [239]. While this low overlap could reflect high false positive rates, it is also possible the effect arises from low coverage or techniques exhibiting biases towards different types of interaction. Other authors have assessed overlap between the *same* affinity purification technique performed by different groups and found only limited overlap between the detected interactions [60, 123, 171, 234]. Similarly, low overlap has also been reported when comparing yeast two hybrid data sets [188]. Again, however, these results may reflect low coverage rather than high false positive rate.

Other quality assessment approaches include comparing PPI data sets to benchmark sets of literature curated interactions, or evaluating the reliability of an interaction through the biological similarity of the interactors, in terms of, for example, correlation in expression patterns or shared biological function. The former approach is extremely sensitive to the choice of benchmark set and is affected by sociological biases in publication and curation processes [236]. Assessing interaction reliability through functional similarity, on the other hand, is dependent on the quality and coverage of functional annotation data, while the use of co-expression assumes interactors are necessarily co-expressed.

In order to circumvent these problems, Venkatesan et al. estimated the precision of yeast two hybrid screens by retesting a random subset of reported interactions using independent interaction assays [236]. This retesting suggested yeast two hybrid screens have a precision of around 80%, which was considerably higher than the precision (approximately 25%) for a literature curated set of interactions retested the same way. Interestingly, when data from TAP-MS screens is retested in a similar way, the performance is much poorer [253]. This difference between the techniques, however, is likely to reflect the difference in the type of interaction (i.e. protein complex co-membership rather than binary interaction) captured by the two techniques, rather than poor data quality from the TAP-MS screens. When the quality of TAP-MS data is assessed through other measures, such as shared biological function of interactors, TAP-MS and yeast two hybrid techniques yield comparable performance [253].

In terms of systematic bias, interaction sets are likely to favour evolutionarily conserved and high abundance proteins [239], although the bias towards highly expressed proteins is less pronounced in yeast two hybrid data. TAP-MS data has also been associated with under-representation of metabolic proteins and over-

representation of proteins involved in transcription and protein synthesis [255]. However, this probably reflects the differing involvement of protein complexes in these cellular functions, rather than bias in the technique itself. Interaction data is also biased against membrane protein complexes because lipid-anchored proteins are hydrophobic and thus more difficult to purify. Recently, affinity purification procedures optimised for membrane proteins have been developed to address this issue [11].

Finally, it is worth noting that the precision and sensitivity of detection techniques are only a partial measure of the usefulness of interaction data for biological inference: interactions captured by *in vitro* assays, even if genuine, do not necessarily have biological relevance. For example, it has been hypothesised that some interaction are evolutionary remnants of past function, but no longer play functional role in the cell [237]. Combining physical protein-protein interaction data with other information capturing functional association (see below) has been suggested as a method for pruning out these 'pseudointeractions.'

### Data Integration

Combining data from multiple screens or sources can be an effective strategy for increasing coverage and reducing noise. Data integration is greatly aided by various public repositories (such as BioGRID [217], IntAct [109], MINT [136], HPRD [23], BIND [4] and DIP [205]) storing interaction data and various databases (for example STRING [101], I2D [23] and iRefIndex [194]) combining these repositories into single datasets. As well as holding information from high-throughput screens, many of these repositories also collate results from smaller-scale studies of protein interaction.

### Dataset Completeness

The mapping of the interactome, the complete set of protein-protein interactions, is still a work in progress. Even the concept of completion is not clearly defined: interactions are likely to be dependent on environmental conditions and cell type [21] and some interactions may, in practice, be undetectable [211]. It is therefore unclear whether the complete interaction should describe the full set of possible interactions [38] or whether maps should be cell type and condition specific.

Estimating how complete our current map of the interactome is difficult because estimating the size of the full interactome is non-trivial. An empirical framework by Venkatesan et al., based on a literature-derived set of high quality true positives and performing repeated screens, gave an estimate of the size of the human interactome of $74000 - 200000$ interactions [236]. Earlier estimates by Stumpf et al. suggested 650000 interactions for the human interactome and 25000–35000 for budding yeast [221]. BioGRID currently holds approximately

150000 unique physical interactions for human and 84000 for yeast and the number of interactions appears to still be growing (Figure 1.5).

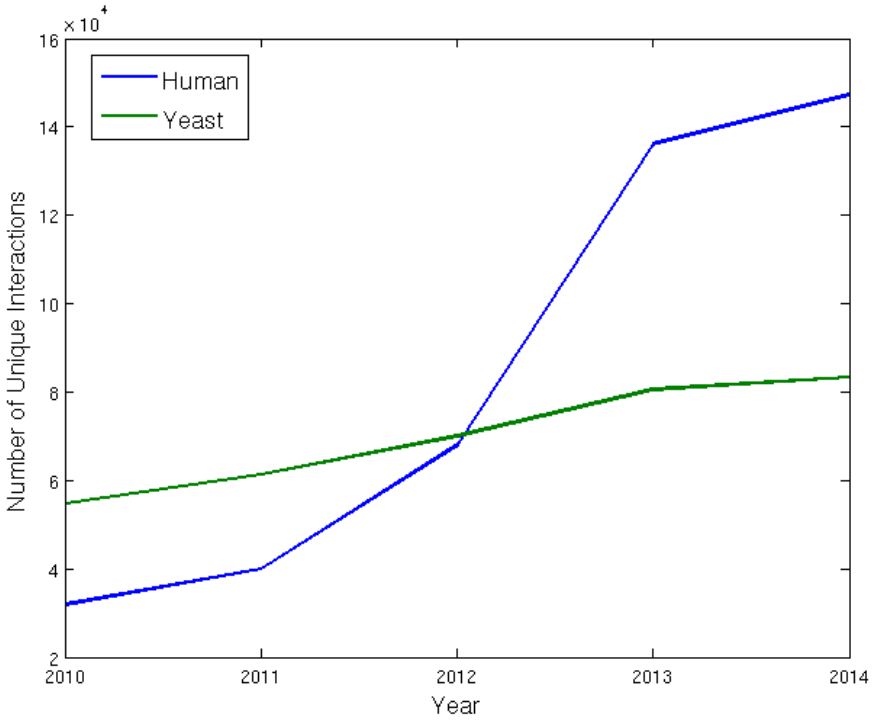


Figure 1.5: The number of unique physical protein-protein interactions in the BioGRID database in July of each year for budding yeast and human.

The incompleteness of the interactome introduces a concern that the network properties of the 'true' interactome may not be the same as those of the mapped sub-network.

Firstly, systematic bias in the detection method may mean that the properties of the sampled sub-network do not reflect the properties of the underlying network. A trivial example would be, for example, under-representation of proteins not expressed in the nucleus in Y2H screens. More subtle effects have also been hypothesised. For example, Caldarelli et al. [26] studied network generation mechanisms where the probability of two nodes being connected is a function of an intrinsic property ('importance') of the nodes. The authors showed it is possible to generate networks with power law degree distributions even in cases where the importance of the nodes is not power law distributed. For example, a network generation algorithm linking nodes if their combined importance exceeded a given threshold gave rise to networks with power law degree distributions even when the importance of the nodes followed an exponential distribution. This has implications for a number of detection techniques: Caldarelli's results suggest that the power law degree distribution observed in protein interaction networks might not reflect the true properties of the network, but instead result from the probability of edge detection being dependent on properties of the proteins. In Y2H screens for example, being able to detect an interaction requires the correct

folding of the reconstructed transcription factor. It is not unrealistic to suggest this may in turn depend on intrinsic properties of the bait and prey proteins, such as size for example, thus leading to a potential distortion of the detected network's degree distribution.

However, not all techniques will be subject to this type of bias. Thus, if the presence of heavy-tailed degree distributions in protein interaction networks were simply an artefact of the nature described by Caldarelli et al., we would expect different behaviour in networks derived with different techniques. A recent study reports that the properties of protein interaction networks are consistent across different detection techniques [99], suggesting these properties are not simply attributable to biases of individual techniques.

A second, more general concern is that even unbiased sampling may lead to distortion in the properties of the detected network. Analytical work by Stumpf et al. showed that, for networks with power law degree distributions, random *node* sampling does not result in the sampled sub-network having the same degree distribution as the original network [222]. Han et al. sampled *edges* from networks with random, exponential, power law, truncated normal degree distribution and found that the resulting sub-networks had similar degree distributions to Y2H-derived partial interactome maps [79]. These results again highlight the concern that incomplete PPI networks are not adequate proxies for the whole interactome and that the properties we observe in the incomplete network are an artefact of the sampling process. It is worth noting, however, that Han's sampled sub-networks did not replicate all properties of the Y2H networks - the clustering coefficient of the sampled networks and real Y2H network, for example, were different [54].

Finally, incomplete coverage is not only a problem when looking at overall graph topology: it can also introduce bias into the network properties of individual nodes. For example, as discussed previously, not all protein interaction data originates from high-throughput screens: online repositories also integrate information from several smaller-scale studies. While this increases coverage, it also biases data towards well studied proteins: it is likely that proteins that are better studied will have more interactions in these datasets, potentially introducing an artificial correlation between degree and features that are of interest to researchers, such as disease association or lethality.

To summarise, results derived from incomplete protein interaction networks may not apply to the full network. There have been some attempts to understand the nature of the bias introduced, but, without knowledge of the topology of the true network, this is a difficult task. The partiality of the coverage therefore needs to be taken into account when interpreting protein interaction networks. Fortunately, coverage of interaction networks is growing steadily. The biases introduced by network incompleteness are therefore likely to diminish with time.

### Predicted Protein-Protein Interactions

Given the noise, bias and limited coverage of experimental interaction mapping techniques, computational approaches for predicting binding partners can offer a valuable complementary perspective. The term predicted protein interaction is used loosely in the literature: either to refer to predicted physical binding or to encompass methods indicative of more general functional association that may or may not involve direct physical contact between proteins [131]. This section will focus exclusively on predicted protein binding. More nebulous types of functional associations will be discussed in Section 1.4.4. In general terms, prediction approaches come in two flavours: biologically motivated methods, which seek to exploit biological insights to predict new interactors and statistical learning methods, which seek to find features which correlate with protein interaction from various types of data, without explicitly requiring knowledge of biology.

There are various biological motivated approaches. The interaction of two proteins depends on their three dimensional features - many prediction methods therefore use protein structure to infer binding partners. For example, some approaches look for pairs of proteins exhibiting commonly interacting protein domains [114], while others seek to 'inherit' interactions from other organisms by identifying interacting pairs of homologs, using either sequence [150] or structure [6, 7] based homology modelling. Recently, more direct methods have also been proposed: Wass et al. used protein docking algorithms, programs traditionally used to predict the structure of complexes formed by known interactors, to detect new interaction partners [245]. While Wass et al. demonstrated this approach is feasible in principle, others have suggested the computational cost of a genome-wide docking-based approach would be prohibitively high [256]. Zhang et al. suggested a less computational intensive approach, based on modelling putative novel interactions on known interactions of structurally similar proteins [256]. Integrating other non-structural information to their prediction method and benchmarking against a set of high confidence interactors, Zhang et al. reported performance generally comparable to, and overall better than, high-throughput experimental methods.

Non-structure based prediction methods also exist. For example, some methods exploit evolutionary relationships between proteins. Because physical interactions occur through the interactions of specific residue interfaces on the proteins [232], interacting proteins are evolutionary linked: the deleterious effect of a mutation perturbing the interaction can be alleviated by a compensating mutation on the other protein. It is therefore possible to predict interaction based on correlated mutations [174] - this principle has lead to a number of methods predicting interactors based on similarity in the evolutionary history (phylogenetic trees) of proteins [37, 206].

Unlike the methods discussed so far, purely statistical approaches make minimal assumptions about the biological mechanisms governing protein interactions

- instead, given a set of 'training examples' (known interactors and (optionally) a set of known non-interactors) and some data about these examples, machine learning methods seek to identify data features that are predictive of the interactions. These methods have the advantage of potentially being capable of exploiting large volumes of heterogeneous data. For example, Pancaldi et al. built a predicted interaction network in fission yeast based on over 100 gene and protein features [168]. The disadvantage is that purely statistical methods are entirely dependent on the quality of the training examples.

Computational methods can also provide a useful tool for prioritising the testing of putative new interactions. However, it is important to note that these methods may be affected by biases in our current understanding of the interactome: computational methods are usually benchmarked against sets of high confidence interactions during development. Systematic biases in these benchmark sets may therefore affect how well computational methods appear to be performing.

### Dynamic and Specific Protein Interaction Networks

Unlike the genome, the interactome is dynamic [21]. Protein expression varies between cell types and during development, meaning the interactome is dependent on both cell type and developmental context. Furthermore, many protein interactions are transient. Thus, even within a specific cellular and developmental context, the interactome is constantly changing. Recent work on PPI networks is seeking to recognise this: while dynamic or condition specific PPI data sets do not yet exist [94], a number of authors have attempted to combine gene expression and protein interaction data to create approximations of dynamic or condition specific networks. Examples of this approach will be discussed in detail in Chapter 5.

Recently, the effects of alternative splicing on protein interactions networks have also received more attention. This is particularly pertinent when working with human networks: current estimates suggest over 60% of human genes undergo alternative splicing [72,149,153]. Both Buljan et al. [25] and Weatheritt et al. [247] found that alternatively spliced regions in the human genome were enriched in conserved protein-protein binding motifs, suggesting alternative splicing may give rise to tissue or cell type specific interactions. Ellis et al. [47] tested this idea experimentally by examining the effect of including or excluding brain specific exons in a number of mouse genes - they found that approximately a third of the alternative splicing events lead to changes in the interactions of the gene products. Davis et al. [40] used a bioinformatic approach to examine the effect of alternative splicing on protein interaction domains, finding evidence for altered interactions in almost 20% of genes. Interestingly, both Buljan et al. and Ellis et al. report that proteins affected by tissue-specific splicing have higher degree and centrality in PPI networks, suggesting alternative splicing is likely to

play a significant role in altering network topology. These results suggest that alternative splicing may fine-tune PPI networks in a tissue-specific manner.

### 1.4.2 Co-Expression Networks

In co-expression networks, edges between genes capture high levels of similarity in expression patterns. The rationale behind the study of these networks is that genes with similar function tend to have similar patterns of expression [46] - co-expression networks therefore provide a perspective on functional associations between genes. The advantage of working with co-expression networks is that many of the concerns raised in relation to PPI network bias and incompleteness are not relevant. Furthermore, condition and cell type specific networks are readily available. On the other hand, the functional significance of co-expression is less clear than that of direct binding and co-expression network generation is associated with its own set of statistical problems.

#### Network Generation

Co-expression networks are conceptually straightforward, but the details of network generation can vary considerably between studies. The most common approach is to use some measure of similarity in expression as a basis for network generation. Various metrics have been proposed. The simplest method is the use of a correlation (either Pearson, see, for example [44] or Spearman see, for example [9]) metric. More sophisticated approaches have been proposed, although it remains unclear whether these offer real benefits. For example mutual information based measures have been used to capture non-linear correlations in gene expression [39]. However, estimating mutual information from expression data can be computational intensive and it remains unclear whether mutual information captures meaningful biological relationships [216]. Networks are normally generated by considering genes with a high correlation magnitude and/or significance value. Here, again, specifics of approaches differ on a number of methodological points: whether absolute values of correlation magnitude are used when thresholding; whether only magnitude or magnitude and significance of the correlation are considered; whether significance values are corrected for multiple testing; whether the resulting network is weighted or unweighted.

A drawback of using correlation-based methods is that they cannot distinguish between direct and indirect dependencies: two genes may be co-expressed because one regulates the other, or because they are both co-regulated by the same transcription factor [148]. Recently, probabilistic graphical models have been suggested as a potential solution to this issue. Probabilistic graphical models use a network representation to encode a probability distribution: nodes represent variables of interest and edges represent conditional dependence. Thus, probabilistic graphical model approaches seek to find the pattern of conditional

dependencies that best explain the gene expression data. These methods include graphical Gaussian models [177, 208] and bayesian network [31, 156] approaches.

Allen et al. [5] performed a comprehensive comparison of different network generation approaches, using both simulated data and real expression data from *E. coli*. Correlation, mutual information and partial correlation based methods all performed comparably in constructing global network topology, with partial correlation based methods being particularly good at identifying few connections with high specificity. Bayesian networks were found to be hindered by their poor scalability to large datasets.

### Co-Expression Data

Until recently, co-expression networks were typically generated from microarray data. Microarrays are a hybridization based technology: sets of one-stranded DNA probes are incubated with fluorescence labelled target sequences. The hybridization of complementary sequences allows inferring the expression levels of sequences corresponding to particular probes. Recently however, progress in transcriptome sequencing (RNA-seq) technologies has allowed the construction of co-expression networks from RNA-seq data. RNA-seq data has the advantage of not having to pre-define the sequences to be measured, not being subject to noise from cross-hybridization and having a greater dynamic range than microarrays [243]. RNA-seq datasets also allow study of novel [200] and alternatively spliced [197] transcripts. On the other hand, there are also concerns relating to RNA-seq data quality: the technique struggles with identification of rare transcripts (as these get obscured by the wide dynamic range) and exhibits a bias towards longer genes (because longer sequences generate more reads) which has not yet been fully addressed by existing normalization methods [225]. It is also worth noting that networks generated from microarray and RNA-seq data may not capture the same functional relationships. In a comparative study of *Arabidopsis* co-expression networks, overlap between RNA-seq and microarray network was low, with microarray networks having higher similarity to known biological networks [68].

### Co-Expression Network Analysis

Topological analysis of co-expression networks has focused, to a large extent, on identifying highly connected nodes and detecting network modules [56]. For example, comparison of network modules and hubs in normal and disease co-expression networks is used to suggest candidate genes for disease association [238]. Guilt-by-association type approaches have also been applied in the context of co-expression results, for example in identifying new players in B-cell signal transduction [16] and plant cell wall synthesis [179].

The analysis and interpretation of the global topological properties of co-expression networks can be confounded by the way they are generated. For

example, correlation is transitive: if A correlates with B and B correlates with C, A and C are also likely to be correlated. The high clustering coefficient of co-expression networks therefore simply reflects this property and cannot be considered indicative of the functional properties of the cell [218]. This also has implications for null model selection: configuration models (networks with the same degree distribution but reshuffled edges) are not necessarily appropriate null models for co-expression networks. Null models generated by computing new networks from permuted versions of the original expression data may therefore be more appropriate under some circumstances.

### 1.4.3 Genetic Interaction Networks

A genetic interaction between two genes refers to the emergence of an unexpected phenotype when variation in the two genes co-occurs. The effect can be negative, for example loss of function in one gene being lethal only when function is also lost in some specific other gene (*synthetic lethality*), as well as positive, for example when the deleterious phenotype of one mutation is rescued by mutation in another gene. These types of interactions are of great interest, because they are thought to play a role in the complexity of biological organisms - for example, in the genetics of complex disease [8].

Genetic interaction networks have been extensively mapped in a number of singled celled organisms, particularly in budding yeast (*Saccharomyces cerevisiae*) [36, 228]. The topology of these networks appears functionally informative in a way reminiscent of PPI networks: similar biological processes cluster together and node degree correlates with functional importance of the node. Genetic interaction networks have also been suggested as tools for identifying potential drug targets [36].

### 1.4.4 Other Functional Association Networks

Protein-protein, co-expression and genetic interaction networks are perhaps the most well studied of gene and protein networks. However, there are a number of other methods of inferring functional association between genes. This section will briefly outline these other forms of interaction.

#### Genomic Context

These methods seek to use genomic information to infer functional associations between genes. One approach is to look at whether gene pairs appear together on multiple genomes: if two gene products need to interact to function correctly, they are more likely to be co-inherited, as loss of one protein would impair the function of the other [176]. We can thus use the correlated absence or presence of gene pairs across multiple genomes ('phylogenetic profile') to infer association. Because of greater availability of sequences genomes in prokaryotes

than eukaryotes, phylogenetic profiling based methods have traditionally been more successful in prokaryotic organisms [191]. Recently, however, Lin et al. proposed an approach supplementing phylogenetic profiles with sequence based information to improve performance in eukaryotes [139]. Other methods base prediction on genomic distance (i.e. how many base pairs separate two genes on the genome): functionally associated genes are known to tend to occur close together in the genome [52]. An extension of this approach is to look for composite proteins, formed from the fusion of two genes into a single gene, to infer a functional association between the two genes in organisms where they remain separate [49]. The STRING database incorporates a number of these methods onto its predicted interaction networks [101].

#### Literature-Based

A number of methods concentrate on retrieving existing information (as opposed to discovering or predicting novel interactions). For example, text mining methods look for statistically significant co-occurrence of gene names in abstracts or articles to infer functional association [20]. Resources collating this data into gene and protein networks include iHOP [87] and STRING. Other approaches use the Gene Ontology (GO), a controlled vocabulary of terms used to functionally annotate genes and gene products, to compute functional similarity between gene pairs based on the number of shared annotations [180].

### 1.4.5 Dynamic Networks

There are important classes of gene and protein networks that are conceptually different from the networks discussed above: metabolic, signalling and gene regulatory networks. Metabolic networks describe the metabolic processes occurring in cells. Details of these representations vary, but the most common approach is to represent metabolites as nodes and enzymes catalysing reactions as edges [103]. Signalling networks depict biochemical events involved in signal transduction within cells such as, for example, phosphorylation cascades. Again, specifics of network construction approaches differ between studies [170]. Finally, gene regulatory networks represent how transcription factors and other transcriptional regulators control gene expression [107].

These classes of networks differ from those described above in a number of significant ways. Firstly, they are directed: the relationships depicted in these networks can be asymmetric. The concept of directionality is trivial in gene regulatory networks, but is also applicable to metabolic and signalling interactions: metabolic reactions can be irreversible, signalling networks generally propagate information in a recognisable direction (usually from outside the cell to the nucleus). Secondly, they are not necessarily genome wide. While for the previously described networks (PPI, co-expression, etc), datasets may not cover the whole genome, the network is theoretically extensible to all genes or proteins.

Metabolic networks are only applicable to enzymes involved in metabolism, signalling networks only to proteins involved in cell signalling and while all genes are regulated by others (i.e. they have in-degree in the gene regulatory network), only a subset of proteins regulate the transcription of others. Thirdly, because of the nature of the relationships depicted in these networks, they are amenable to dynamic modelling. Various levels of representation have been employed, from logical models [70] to systems of differential equations [117]

These smaller-scale dynamic network approaches are a valuable complementary approach to the larger-scale static network representations. Some of the tools applied to the analysis of static network representation have also been applied to these dynamic networks, such as, for example, topological analysis of metabolic networks [73], along with other structural approaches, such as stoichiometric analysis [182]. However, it is important to recognise that because of the differences between static and dynamic, and genome-wide and specialised networks, the same approaches may not always be optimal for both.

There has also been interest in integrating these networks with other gene and protein network representations, for example for predicting the phenotypic consequences of genomic variation [111]. How to best deal with the differences in scope and edge type of these networks when integrating them with other gene and protein networks remains an open question.

## 1.5 Thesis Overview

In summary, while network approaches show promise as a systems-level approach to cellular biology, significant challenges remain. How to best extract biologically interesting information from network representations remains unclear. This is the central question explored in this thesis: I will discuss the development and application of network tools in three biological scenarios:

- Chapter 2 looks at prediction of protein function using functional association networks, developing novel prediction methods as well as explicitly addressing the issues involved in the benchmarking of prediction algorithms. (The work in Chapter 2 was co-supervised by John Shawe-Taylor).

- Chapters 3 and 4 address the question of how genetic variation gives rise to variation in phenotype. Specifically, we will be looking at networks methods to understand loss of function variation in the human genome. (This work was partially undertaken as a visiting scholar in Mark Gerstein's laboratory).

- Chapters 5 and 6 model changes in cellular state in terms of changes to protein interaction and gene co-expression networks. In particular, we will be applying network approaches to model cell response to oxidative

stress. The work presented in Chapter 5 has been published in Lehtinen et al. [133].

# Chapter 2

# A Graph Kernel Method for Gene Function Prediction

## 2.1 Introduction

### 2.1.1 Gene Function Prediction: Scope and Definitions

Modern biology is characterized by rapidly increasing volumes of genomic and proteomic data. As a consequence, there is much interest in automated extraction of functionally meaningful information from these datasets. One key approach is the *in silico* prediction of gene and protein function.

Gene and protein function prediction are terms that encompass a large variety of problems and approaches. The interpretation of the term *function* is broad and covers different levels of abstraction: definitions range from a protein's biochemical role to its impact on phenotype. The aspect of function considered depends on the data set at hand and the biological context of the prediction.

Depending on the context of the problem, prediction can be approached from two different angles. *Gene-centric* approaches aim to predict what function a gene might be involved in while *function-centric* approaches focus on predicting novel genes involved in a particular function. Although this chapter will focus mainly on function-centric methods, it is worth noting these two approaches are closely related and, in many contexts, are considered interchangeable.

Owing to the scope of the problem, a variety of data sources and prediction methods have been exploited in gene function prediction. In general terms, prediction methods fall into two broad categories: *de novo* methods seeking to predict function based on intrinsic properties of a gene and *guilt-by-association* (GBA) approaches, which predict new functional labels based on a gene's similarity to already functionally characterised genes.

A number of established GBA-type prediction methods base their predictions on sequence or structural similarity. Recently however, in response to the increasing prevalence of functional association data, there has been considerable interest in developing GBA methods exploiting functional association networks.

This chapter explores network-based function prediction using *kernel methods* and explicitly addresses difficulties associated with the benchmarking of GBA methods.

### 2.1.2 Exploiting Networks for Function Prediction

Early uses of functional association networks for function prediction focused on local network connectivity, predicting a gene's function based on the function of its direct neighbours [85, 212] or its 2-neighbourhood (neighbours' neighbours) [32]. While these approaches showed promise, they failed to take advantage of the wider network topology, which has been shown to contain meaningful information about the functional clustering of genes [248].

Meanwhile, other approaches have focused solely on finding clusters in the whole network. For example, functional prediction has been approached as a graph partition task: methods have aimed to allocate nodes into functional categories in a way that minimizes the number of interactions between the categories [106, 235]. These approaches have been criticised for not fully exploiting network proximity information [191]: all genes within a functional category are considered equally functionally associated, regardless of the distance between them in the network. Depending on the problem at hand, this may be unhelpful - for example, if we are interested in prioritizing candidate genes for experimental validation.

As both global topology and local proximity are predictive of functional association, there has been considerable interest in developing methods exploiting both these features. Although the specifics of these methods vary, fundamentally, they all seek to build on the same idea: that the functional similarity between genes relates to how reachable one gene is from the other. In other words, the functional similarity of two genes depends on 1) how close the genes are in the network and 2) how many paths connect the two. Thus, it is perhaps not surprising many of the network-based methods can be expressed in terms relating to 'walks' on the network. This idea will be explored further in the Technical Background (Section 2.2). There are two main classes of methods exploiting both global topology and local proximity: probabilistic network models and kernel methods.

A number of authors have implemented probabilistic network models for gene function prediction. A probabilistic network model is a mathematical construct representing dependencies between random variables. In the context of gene networks, these models capture how a gene's function depends on that of its network neighbours. A number of approaches have modelled the problem in terms of belief propagation in these networks [42, 121, 134, 231]. GeneMANIA [244], one of the most successful prediction algorithms to date [155, 175], makes use of this approach, implementing Gaussian label propagation (more details on the GeneMANIA algorithm will be provided in Section 2.5.2). To our knowledge,

no prediction algorithm has out performed GeneMANIA. The performance of the prediction methods developed in this chapter will therefore be benchmarked against GeneMANIA.

The other major class of methods makes use of kernels. Kernel approaches transform functional association networks into functional similarity scores between genes, based on the topology of the network (as discussed further in Section 2.2). This representation allows the use of statistical learning approaches (for example regressions) on network data. Existing methods have most commonly used diffusion kernels, paired with support vector machines [127] or logistic regression [130]. A related method, FunctionalFlow [157], makes use of a diffusion kernel-like process.

While most existing methods have focused on diffusion kernels, recently, work by Heriche et al. compared different kernel functions (i.e. different ways of generating similarity scores between genes from the network) [82]. In this work, the *commute time kernel* was found to perform most robustly: when tested on a number of different benchmarks, this kernel was consistently among the top performers, while other kernel's performance fluctuated significantly. Furthermore, the authors argue that the performance of most other kernels is dependent on correct parametrization, which, as discussed below, can be problematic. The commute time kernel, on the other hand, is parameter free.

In Heriche et al.'s work, kernels are exploited for prediction using a nearest neighbour approach (see Technical Background, Section 2.2, for a detailed overview). More complex algorithms for kernel based prediction have been well documented [213]. The performance of commute time kernels paired with more complex prediction algorithms has not been explored.

### 2.1.3 Problems with Network Based Approaches

Despite the widespread interest, network-based prediction approaches are not unproblematic. The central concern is that our validation paradigms are unable to distinguish between methods which reliably detect patterns which will allow us to predict new annotations and methods which simply capture features of existing data. This section will discuss potential problems with validation paradigms and network-based approaches themselves.

#### Cross-Validation

Accurate evaluation of the performance of prediction methods is essential for meaningful comparison of different algorithms. This requires sets of genes known to be functionally associated to use as examples of true positives. These known labels are commonly derived from the Gene Ontology (GO) [10].

The GO was originally developed to provide a controlled vocabulary of terms relating to the biological function of genes and gene products. The GO labels genes with standardised descriptions of functions, relating to one of the three

main 'branches' of the ontology (biological process, molecular function and cellular component). Terms are organized in a hierarchical manner, allowing descriptions at different levels of specificity. For benchmarking purposes, authors consider genes labelled with the same term as a 'set' sharing the same function.

A typical benchmarking approach is cross-validation: a subset of known labels are hidden, and the performance of the method is assessed by how well the hidden labels are recovered. A concern with using cross-validation to compare prediction algorithms is that performance is assessed with respect to labels which are already known, while we are actually interested in the ability to predict new labels. If the properties of undiscovered labels are not the same as those of the known labels, cross-validation may not be a reliable indicator of real performance.

This is particularly pertinent in the context of protein function prediction: the propagation of information across biological databases raises concerns about the similarity of known and undiscovered labels [131]. The discovery of new functional labels may affect the content of functional association networks and vice versa - thus potentially leading to differences in the way known and undiscovered labels are represented in functional association datasets. For example, the discovery of new functional associations between genes affects the labelling of proteins in databases such as GO or KEGG (kyoto encyclopedia of genes and genomes [105]). If we then use cross-validation on a GO or KEGG dataset to assess a prediction method, the results may not actually reflect the algorithm's ability to predict function for new genes, but rather the extent to which information has been dissipated across databases (Figure 2.1).

Recent work has explicitly investigated this problem by looking at the GO annotations of genes which interact in PPI networks. 13% of GO annotations shared by interacting genes were found to be derived from the same publication that reported the interaction [66], confirming the idea that information does indeed propagate between databases. Furthermore, the authors found a low (r=0.2) but significant correlation between how well guilt-by-association methods perform for a particular term (as assessed by cross-validation) and the extent of this overlap between network and gene annotation data.

Interestingly, similar problems have also been reported for sequence similarity based prediction algorithms. The GO derives some of its annotations from sequence similarity (for example 'IEA' (inferred from electronic annotation, 'ISS' (inferred from sequence similarity)). Again, this raises the concern that the dataset used for evaluation is not independent from the dataset used for prediction, potentially leading to a biased estimation of predictive performance. Indeed, Rogers and Ben-Hur [201] showed that including these evidence codes when benchmarking a prediction algorithm tends to over-estimate how well sequence similarity based methods perform.

These problems also raise the issue of method parametrization. Parameter

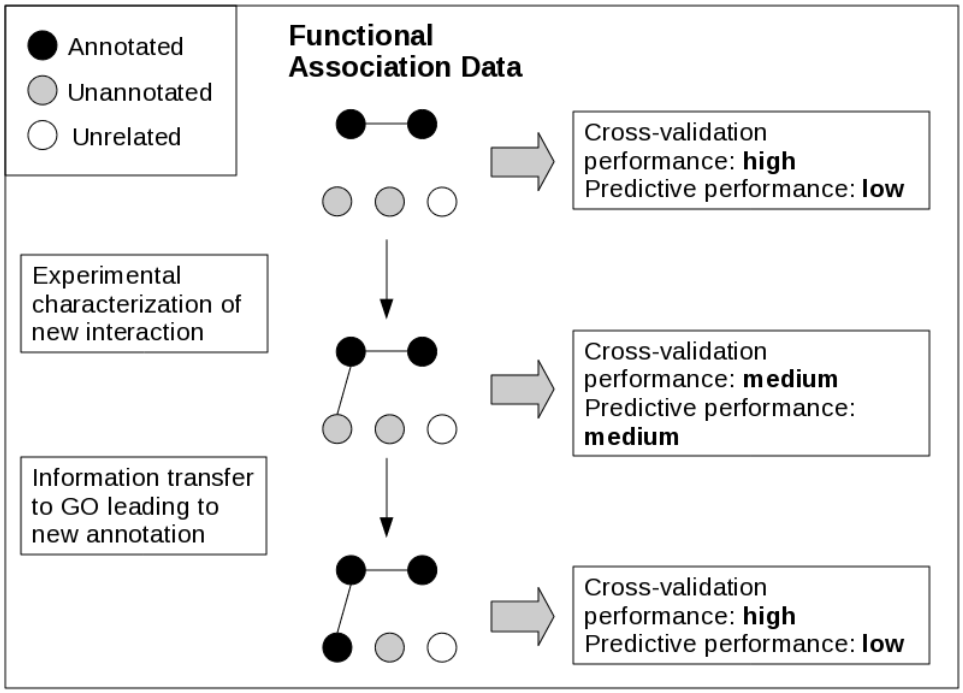


Figure 2.1: A schematic illustration of how the transfer of information from prediction data to benchmarking data can interfere with assessing the predictive power of algorithms. Dark circles represent genes annotated with a particular function, light circles represent unannotated genes. Characterisation of new functional associations (for example protein-protein interactions from a yeast two hybrid screen or the identification of new binding partners from a smaller scale study) leads to new GO annotations being made. As a result, an association based prediction method that seems to perform well under cross-validation may actually be a poor predictor of new annotations: the good performance simply reflects the non-independence of the functional association and annotation data.

choices can only be made based on what is known at the time of prediction. However, this will optimize performance on the set of already known labels, which, as discussed, may not actually be the optimal setting for discovering new labels (a problem akin to over-fitting). The extent to which this affects prediction performance has not been investigated.

### More Realistic Benchmarks

There have been significant efforts to compare prediction algorithms using a more realistic benchmark. Competitions such as CAFA (Critical Assessment of Function Annotation) [187] and MouseFunc [175] evaluate prediction methods based on novel true positives uncovered after the predictions have been made. Thus, unlike cross-validation, this benchmark directly assesses an algorithms' ability to predict novel annotations.

Despite being welcomed as an attempt to provide fair comparative assessment of methods, CAFA-style competitions have also attracted criticism, particularly

because of their use of GO annotation.

A major concern is that the process of label acquisition is affected by existing annotations, extending the problems with cross-validation to benchmarks based on new labels as well. This problem has mainly been raised in the context of sequence similarity based methods. As discussed previously, genes may have 'IEA' annotations for a particular function based on high sequence similarity with better annotated genes. While these annotations are ignored in the CAFA challenge, they may have an indirect effect on the assessment: if the presence of IEA annotations makes it more likely for the gene to acquire a high confidence annotation for the same label (for example, through GO curation or by guiding the direction experimentation), methods predicting new labels based on sequence similarity will appear to perform well because they replicate this feature of label acquisition.

Gillis and Pavlidis [67] explicitly addressed this concern by showing that simply using pre-existing 'IEA' annotations as a predictor of future high confidence annotations performed comparatively to the best CAFA entries in the 2012-2013 competition. This suggests that (sequence based) computational methods may simply be re-creating the 'IEA' annotation and therefore seem to perform well, not because of actual predictive power, but because they mimic the process of annotations becoming incorporated in the GO.

Gillis and Pavlidis discussed this problem in the context of sequence based prediction. However, we hypothesise that similar concerns may also be relevant for network based prediction if the incorporation of annotations into the GO is affected by existing functional annotation data.

A second concern is that CAFA style benchmarks cannot differentiate between true negatives (i.e. genes that do not have a particular function) and undiscovered labels [43]. This is particularly problematic in light of the biases highlighted previously: if these biases exist, CAFA style competitions run the risk of undervaluing methods that provide genuine insight, in favour of methods that simply mimic the process of GO annotation acquisition.

### Predictions Biases

Aside from problems relating to benchmarking, there are also concerns relating to biases inherent in network-based approaches.

Gillis and Pavlidis argue that network-based methods simply predict more labels for already well characterized nodes ('rich get richer') instead of providing function specific insight [64]. The authors showed that ranking genes based on multifunctionality (a measure relating to the number of GO categories a gene is annotated with) and then using this single ranking to predict membership in *multiple* GO categories yields a very high average performance. A relatively good, though weaker, performance is also achieved by ranking genes by network degree. There is a moderate correlation between a gene's degree and its multi-

functionality, although the strength of this correlation varied between networks. Crucially, the magnitude of this correlation in different networks correlates with the performance of the degree based ranking, suggesting the functionally meaningful information captured by degree is actually information about multifunctionality. Furthermore, the authors found that although network-based methods outperformed the degree based ranking, the performance of the two methods was highly correlated. The authors thus suggest that network based prediction is highly dominated by a gene's degree, which in turn simply reflects the gene's multifuctionality. Thus, the authors argue that network based methods are simply predicting more labels to genes that are already well characterized, but, crucially, do not provide *function specific* insight.

Extending this work on multifunctionality, the authors find that a small number of edges between highly multifunctional genes encode much of the functional information in the network [65], raising the concern that GBA properties are not generalizable to the network as a whole.

The origin of this multifunctionality effect is unclear. One possibility is that it reflects a genuine property of biological networks: highly connected genes may indeed be more likely to be highly multifunctional. On the other hand, the effect may also be due to biases in the way genes are annotated and interaction data acquired. Regardless of the cause, the lack of function specific insight is clearly an undesirable property for a function predictor.

### 2.1.4 Aims and Objectives

This Chapter will explore a new method for function prediction, based on using a kernel combined with a dimensionality reduction approach ('Compass'). We will evaluate the performance of Compass against the GeneMANIA algorithm on a number of benchmarks. We develop a 'GO rollback' benchmark to mimic CAFA-style prediction competition and two benchmarks (RNAi and ageing) based on experimental screens linking genes to particular phenotypes. We will use these these benchmarks to explicitly investigate the biases in prediction and prediction evaluation discussed in the Introduction.

## 2.2 Technical Background

This section provides technical background on relevant tools used in network based gene function prediction.

### 2.2.1 Kernel Methods

A kernel is a function that gives the inner product of two vectors in a multidimensional space (referred to as a feature space):

$$K(x, y) = \langle \phi(x), \phi(y) \rangle$$

where $\phi(x)$ is the mapping of $x$ onto the feature space and $\langle \cdot, \cdot \rangle$ takes the inner product. The matrix of inner products $K$ is referred to as the *kernel matrix*.

To build an intuition for kernel functions, we can think of them as generating a measure of similarity between two data points in a particular feature space. Thus, different kernels, mapping to different feature spaces, represent different notions of similarity.

In general, the motivation behind kernel methods is the hope that mapping data into the feature space will aid pattern detection - for example, as illustrated in Figure 2.2, making the data linearly separable.

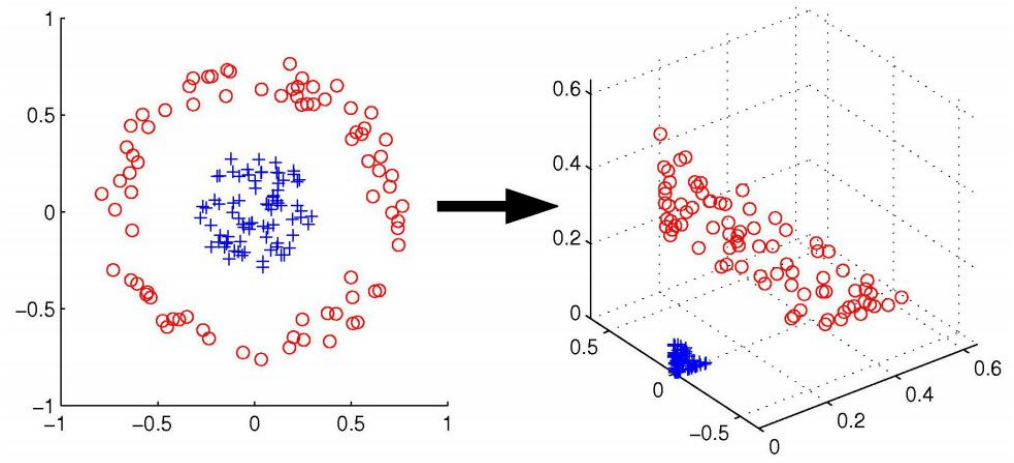


Figure 2.2: Illustration of how a mapping into a different space can make patterns in data more detectable. The two figures show the same data in two different spaces. In the original space, the non-linear boundary between the two groups (red circles and blue crosses) is detectable by eye, but difficult to handle using linear methods. Mapping the data into a three dimensional feature space using quadratic map $\phi(x, y) = (x^2, y^2, 2\sqrt{2}xy)$ makes the groups linearly separable. Adapted from stackexchange.com

The usefulness of kernel methods lies in avoiding having to explicitly compute the mapping: a number of statistical learning algorithms, such as support vector machines, linear regression and principal component analysis, can be applied to the kernel matrix instead of the original data. We can often generate the kernel matrix without explicitly mapping into the features space. In many contexts, this is useful because the mapping is computationally expensive.

In the context of guilt-by-association prediction, however, the motivation for applying kernel learning is slightly different. Unlike a standard learning problem, our data is not in the form of a set of feature vectors, we only have access to a network representation. Therefore, transforming the network into a kernel matrix allows us to apply a broad range of statistical learning methods to network data.

Another advantage of kernel methods in the context of guilt-by-association prediction is their suitability to problems involving data integration. Protein function prediction often exploits information from a variety of heterogeneous

sources. Kernels have a number of properties which make them well suited for this type of problem.

Firstly, any symmetric, positive semi-definite function is a valid kernel [213]. This means that any symmetric matrix with non-negative eigenvalues is a valid kernel matrix - i.e. the representation of inner products of a set of vectors in some feature space. Therefore, we can define a similarity measure and determine whether it is a valid kernel simply through analysis of the similarity matrix. We do not necessarily need to know anything about the feature space the similarity measure relates to. This makes kernel methods applicable to data types much more diverse than vectors. Thus, kernels allow us to represent heterogeneous data in a common format.

Secondly, various mathematical operations (including addition, multiplication and exponentiation) preserve positive semi-definiteness, meaning we can, for example, simply add kernel matrices in order to combine them and still have a valid kernel matrix.

### Graph Kernels

The usefulness of kernel methods for guilt-by-association prediction is crucially dependent on the choice of kernel: the representation is only meaningful if the position of points in the feature space reflects functional similarity. In a network where edges represent evidence of functional association, how functionally similar two nodes are, intuitively, depends on 1) the proximity of the two nodes (i.e length of shortest path between the nodes) and 2) the density of connectivity between the two nodes (the number of paths between the two nodes). A number of similarity measures seek to capture this idea using the idea of a *random walk*, a stochastic process where a 'walker' transitions from node $i$ to $j$ with probability $w_{ij}/d_i$, where $w_{ij}$ is the weight of the edge between nodes $i$ and $j$ and $d_i$ is the weighted degree of node $i$ ($d_i = \sum_j w_{ij}$).

Below we will discuss three commonly used kernels which can all be interpreted in terms of a random walk on a network. First, however, we will introduce the idea of the *graph Laplacian* $L$. Similarly to the adjacency matrix $A$, the graph Laplacian is a matrix representation of a network: $L = D - A$, where $D$ is a diagonal matrix containing the degree of each node. The Laplacian matrix has a number of useful properties and is key to the computation of the kernel matrices we are interested in.

**The Diffusion Kernel**: $K_D = e^{\beta L} = \lim_{n\to\infty}(1 + \frac{\beta L}{n})^n$,

where $L$ is the graph Laplacian and $\beta$ is a constant. The diffusion kernel captures a measure of similarity between nodes relating to a lazy random walk (a variant of a random walk where the walker remains in place with probability $1 - d_i\beta$). $K_D(i, j)$ represents the sum of the probabilities the walker will take each of the paths between nodes $i$ and $j$ [120].

**The Commute Time Kernel**: $K_{CT} = L^+$,

where $L+$ is the Moore-Penrose pseudoinverse of the Laplacian, which satisfies the following:

$$LL^+L = L$$
$$L^+LL^+ = L^+$$
$$(L^+L)^H = L^+L$$
$$(LL^+)^H = LL^+$$

where $X^H$ is the conjugate transpose of $X$.

The commute time kernel is a representation of the data in a space where genes are separated by the average number of steps it takes for a walker to transition between two nodes [53, 186].

**Random-Forest Kernel**: $K_{RF} = (I + L)^{-1}$

In the context of a random walk on a graph, $K_{RF}$ relates to the probability of transitioning between two nodes in a random walk with a random number of steps [28].

### 2.2.2 Kernelized Prediction Algorithms

GBA protein function prediction is a supervised learning problem: we seek to predict new labels based on a training set of known labels (or 'examples'). These known labels are represented as a vector $\mathbf{y}$, where $y_i = 1$ if gene $i$ is involved in the function, $-1$ if known *not* to participate in the function and 0 if unlabelled (see below for further discussion of negative labels). Our aim is compute a score vector $\hat{\mathbf{y}}$ representing the likelihood of genes participating in the function. There are a number of ways graph kernels can be exploited for this type of prediction.

#### Nearest Neighbour Method

The simplest approach, explored by Heriche et al. [82] is to treat the kernel as a look up table: each gene is simply assigned a score based on the sum of its similarities with known pathway members:

$$\hat{\mathbf{y}} = \mathbf{K} \cdot \mathbf{y}$$

This approach treats the kernel matrix as a table of similarities between genes, without making use of the fact that the similarities actually represent inner products between gene vectors in the feature space. It is possible to express a number of statistical learning algorithms in forms where they can be applied to these inner products instead of the feature vectors themselves. We will present a brief over-view of some of the major algorithms. Throughout this section, vectors and matrices will be in bold, to distinguish them from scalars.

### Regression Methods

Regression method can be applied to data in kernel form. Regression algorithms seek to find a model relating the target vector $\mathbf{y}$ to the feature vectors (often referred to as independent variables in the context regression algorithms) $\mathbf{X}$. This model can then be used to compute $\hat{\mathbf{y}}$ for unlabelled genes. To illustrate how these algorithms can be applied to kernel data, we will use the example of a ridge regression.

A linear regression models the relationship between the target vector and feature vectors as a linear combination of the feature vectors:

$$\mathbf{y} = \mathbf{X}\mathbf{W} + \boldsymbol{\epsilon}$$

where $\mathbf{X}$ is a matrix in which rows are the feature vectors, $\mathbf{W}$ is the weighting of each feature (i.e dimension of the feature vector) and $\epsilon$ is a random error term. We thus want to choose $\mathbf{W}$ to minimize the error term (or 'loss function') - specifically, we are interested in minimizing the sum of the squares of the error terms. This is known as the *least squares approximation*:

$$||\boldsymbol{\epsilon}||^2 = (\mathbf{y} - \mathbf{X}\mathbf{W})'(\mathbf{y} - \mathbf{X}\mathbf{W})$$

The optimization problem can then be solved by setting the derivative of the loss function (with respect to $\mathbf{W}$) to zero. Provided the inverse of $\mathbf{X}'\mathbf{X}$ exists, this solution can be expressed in terms of a linear combination of the training points:

$$\mathbf{W} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-2}\mathbf{X}'\mathbf{y} = \mathbf{X}'\boldsymbol{\alpha}$$

where $\boldsymbol{\alpha} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-2}\mathbf{X}'\mathbf{y}$

It is entirely possible that the inverse of $\mathbf{X}'\mathbf{X}$ does not exist. These cases correspond to *ill posed* problems: there is not enough information to precisely specify the solution. One way to solve this problem is to add additional constraints - this is known as *regularization*. A natural constraint is to favour simpler models by including the norm of the weight vector into the loss function: instead of minimizing $||\boldsymbol{\epsilon}||^2$ we now want to minimize $||\boldsymbol{\epsilon}||^2 + \lambda||\mathbf{W}||^2$, where $\lambda$ is a positive constant controlling the relative importance of the two constraints.

The solution to the optimization problem then becomes:

$$\mathbf{W} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y}$$

where $\mathbf{I}$ is the identity and thus $(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})$ is always invertible. Rearranging to obtain an expression in terms of a linear combination of the training points:

$$(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})\mathbf{W} = \mathbf{X}'\mathbf{y}$$
$$\mathbf{W} = \lambda^{-1}\mathbf{X}'(\mathbf{y} - \mathbf{X}\mathbf{W}) = \mathbf{X}'\boldsymbol{\alpha}$$

where $\boldsymbol{\alpha} = \lambda^{-1}(\mathbf{y} - \mathbf{XW})$. The expression for $\boldsymbol{\alpha}$ can be re-arranged to give:

$$\boldsymbol{\alpha} = (\mathbf{XX}' + \lambda\mathbf{I})^{-1}\mathbf{y} = (\mathbf{K} + \lambda\mathbf{I})^{-1}\mathbf{y}$$

where $\mathbf{K}$ is the product of inner products of the feature vectors (i.e. kernel matrix). Thus, we have expressed the ridge regression in a form requiring only the target vector $\mathbf{y}$ and the kernel matrix.

**Dimensionality Reduction**

Dimensionality reduction methods aim to find lower dimensional representations of datasets, either by recoding the data to reduce dependence between dimensions (a technique known as *principal component analysis* or PCA) or by selecting dimensions that are more relevant for the predictions task (known as *partial least squares* or PLS).

Principal component analysis aims to re-express a dataset in terms of uncorrelated dimensions (*principal components*), choosing these dimensions to retain as much of the variance in the original data as possible. Thus, the problem of choosing the directions of maximal variance can be expressed as choosing $\mathbf{w}$ so as to maximise $\mathbf{w}'\mathbf{Cw}$, subject to $||\mathbf{w}|| = 1$, where $\mathbf{C}$ is the covariance matrix given by $\mathbf{C} = \frac{1}{l}\mathbf{X}'\mathbf{X}$ and $l$ is the number of data points. This definition of the covariance matrix assumes the data is centred (i.e. has a mean of zero). It is worth noting however, that the work presented below in deriving kernel versions of PCA and PLS does not make use of this assumption and is valid even with non-centred data.

This optimization problem is solved by choosing the directions as the eigenvectors of $\mathbf{C}$. Hence, the projection of the data onto the $j^{th}$ principal component is given by $(\mathbf{Xu_j})$, where $\mathbf{u_j}$ is the eigenvector corresponding to the $j^{th}$ largest eigenvalue. Choosing the $m$ first eigenvectors thus gives a projection onto an m-dimensional subspace. The number of dimensions is usually chosen depending on how much of the variance we want to preserve.

We can also compute this projection using the kernel matrix. There is a relationship between the eigenvectors of $\mathbf{C}$ and $\mathbf{K}$. If $\mathbf{v}$ and $\lambda$ are an (normalized) eigenvector-eigenvalue pair of $\mathbf{K}$, then:

$$l\mathbf{CX}'\mathbf{v} = \mathbf{X}'\mathbf{XX}'\mathbf{v} = \mathbf{X}'\mathbf{Kv} = \mathbf{X}'\mathbf{v}$$

meaning $\mathbf{X}'\mathbf{v}$ and *lambda* is an eigenvector-eigenvalue pair for $l\mathbf{C}$. Furthermore, $||\mathbf{X}'\mathbf{v}||^2 = \mathbf{v}'\mathbf{XX}'\mathbf{v} = \lambda$. Thus, the corresponding normalized eigenvector of $l\mathbf{C}$ is $\mathbf{u} = \lambda^{-\frac{1}{2}}\mathbf{X}'\mathbf{v}$. Hence, the projection on to the $j^{th}$ principal component is:

$$\lambda^{-\frac{1}{2}}\mathbf{Xu_j} = \lambda^{-1}\mathbf{XX}'\mathbf{v_j} = \lambda^{-1}\mathbf{Kv_j} = \mathbf{v_j}$$

.

Thus, we can compute the projection onto the principal components using only the principal components of the kernel matrix.

For regression type problems, we can then perform a standard regression in the new subspace. The advantage of this approach is that by removing the directions of low variance, we are potentially de-noising the data. The disadvantage, however, is that we are not necessarily choosing the directions which are most predictive. Partial least squares regression seeks to address this problem, by choosing directions maximizing the *covariance* between the feature vectors and target.

Theoretically, the directions maximizing covariance between $\mathbf{X}$ and $\mathbf{y}$ can be found using a singular value decomposition (svd) of the covariance matrix $l\mathbf{C_{xy}} = \mathbf{X'y}$. However, this approach is restricted in the number of directions which can be extracted: the number of non-zero singular values of $\mathbf{X'y}$ is restricted by the number of dimensions of $\mathbf{y}$.

Instead, PLS chooses only the first direction in this manner. Then, instead of choosing further directions as orthogonal to $\mathbf{u_1}$ as we would with the svd, we instead look for directions orthogonal to the projection used in the regression i.e. $\mathbf{Xu_1}$. We can find the next direction by projecting $\mathbf{X}$ into the space orthogonal to $\mathbf{Xu_1}$, a process referred to as 'deflating' $\mathbf{X}$. Thus, with $\mathbf{X_1} = \mathbf{X}$, we obtain $\mathbf{X_2}$ as:

$$\mathbf{X_2} = (\mathbf{I} - \frac{\mathbf{X_1 u_1}(\mathbf{X_1 u_1})'}{(\mathbf{X_1 u_1})'\mathbf{X_1 u_1}})\mathbf{X_1}$$

and then recomputing $\mathbf{C_{xy}}$ and performing svd again, iterating until the desired number of directions have been found. Deflating $\mathbf{y}$ is redundant: removing explained covariance has no effect on the extraction of subsequent directions.

Once the desired number of directions have been found, we then perform a regression using the projection of data onto these directions. For the PLS regression, it is possible to express $\mathbf{W}$, the matrix of regression coefficients as (see references [145, 190] for derivation):

$$\mathbf{W} = \mathbf{U}(\mathbf{P'U})^{-\mathbf{1}}\mathbf{C'}$$

where $\mathbf{U}$ is a matrix with columns $u_j$, the direction found on the $j^{th}$ iteration, $P$ is a matrix with columns $\mathbf{p_j} = \frac{\mathbf{X_j' X_j u_j}}{\mathbf{u_j' X_j' X_j u_j}}$ and $\mathbf{C}$ has columns $\mathbf{c_j} = \frac{\mathbf{y' X_j u_j}}{\mathbf{u_j' X_j' X_j u_j}}$, where $\mathbf{X_j}$ corresponds to the $j^{th}$ deflation of the $\mathbf{X}$.

As before, we now seek to express this process in terms of the kernel matrix $\mathbf{K}$. To achieve this, we re-express $\mathbf{u_j}$ as $a_j\mathbf{u_j} = \mathbf{X_j'B_j}$. By definition, if $\mathbf{v_j}$ and $\sigma_j$ are the right singular vector and the singular value corresponding to $\mathbf{u_j}$ and we

use $\mathbf{y_j}$ to indicate the $j^{th}$ deflation of $\mathbf{y}$:

$$(\mathbf{X'_j y_j})\mathbf{v_j} = \sigma_\mathbf{j}\mathbf{u_j}$$
$$(\mathbf{X'_j y_j})'\mathbf{u_j} = \sigma_\mathbf{j}\mathbf{v_j}$$
$$\rightarrow \mathbf{X'_j y_j y'_j X_j u_j} = \sigma_j^2\mathbf{u_j}$$

This leads to the following recursion: $\mathbf{B_j} = \mathbf{y_j y'_j X_j X'_j B_j} = \mathbf{y_j y'_j K_j B_j}$, with the normalization $\mathbf{B_j} = \frac{\mathbf{B_j}}{||\mathbf{B_j}||}$. This allows us to estimate $\mathbf{B_j}$ without needing $\mathbf{X_j}$. To compute the deflation, we define $\mathbf{t_j} = a_j\mathbf{X u_j}$:

$$\mathbf{t_j} = a_j\mathbf{X_j u_j} = \mathbf{X_j X'_j B_j} = \mathbf{K_j B_j}$$

The deflation of the kernel can then be computed as:

$$\mathbf{K_{j+1}} = \mathbf{X_{j+1} X'_{j+1}}$$
$$= (\mathbf{I} - \frac{(\mathbf{X_j u_j})(\mathbf{X_j u_j})'}{(\mathbf{X_j u_j})'(\mathbf{X_j u_1})})\mathbf{X_j X'_j}(\mathbf{I} - \frac{(\mathbf{X_j u_1})(\mathbf{X_j u_1})'}{(\mathbf{X_j u_1})'(\mathbf{X_j u_1})})$$
$$= (\mathbf{I} - \frac{\mathbf{t_j t'_j}}{\mathbf{t'_j t_j}})\mathbf{K_j}(\mathbf{I} - \frac{\mathbf{t_j t'_j}}{\mathbf{t'_j t_j}})$$

For computing the regression coefficients in terms of the kernel matrix, we express these as $\mathbf{W} = \mathbf{X'}\boldsymbol{\alpha}$ so that $\hat{\mathbf{y}} = \mathbf{K}\boldsymbol{\alpha}$. By expressing $\mathbf{W} = \mathbf{U}(\mathbf{P'U})^{-1}\mathbf{C'}$ in terms of $\mathbf{B}$ and $\mathbf{K}$, we get the following expression for $\boldsymbol{\alpha}$ (for details of the derivation, refer to Shawe-Taylor and Cristianini [213]):

$$\boldsymbol{\alpha} = \mathbf{B}(\mathbf{T'KB})^{-1}\mathbf{T'y}$$

where $T$ is a matrix with columns $t_j$.

In summary, both PCA and PLS regressions can be expressed in forms that only require us to know the kernel matrix.

### Support Vector Machines

Support vector machines (SVMs) are another set of algorithms often used in the context of kernel learning. SVMs aim to find a hyperplane that separates points labelled $y_i = 1$ and $y_i = -1$ so that the distance between the hyperplane and the nearest point from each category is maximized. Unlabelled genes are then categorized based on which side of the hyperplane they fall. Alternatively, $\hat{\mathbf{y}}$ can be computed as the distance of the unlabelled genes from the hyperplane. Like regression and dimensionality reduction algorithms, SVMs can be expressed in both primal and dual forms.

### 2.2.3 Protein Function Prediction and Negative Examples

So far in this section, we have discussed prediction methods in general terms, without considering gene function prediction specifically. However, it is worth briefly discussing the selection of training examples in the context of gene function. The choice of positive examples is generally not problematic: these could be, for example, a specific genes set of interest or genes known to be involved in some function (for example, a particular GO term).

Some prediction methods, however, also require negative examples, that is, a set of genes known *not* to participate in a function. The selection of negative examples is more challenging: any bias in the choice of these genes will introduce a bias in prediction. Efforts have been made to produce systematic negative gene sets for GO terms. While these have been reported to improve performance in some context, they can also have a detrimental effect [251]. Furthermore, for functions of interest that do not correspond to GO terms, negative sets are rarely available. Alternatives to choosing a set of negative examples include treating all unlabeled genes as negatives in the training phase or randomly sampling the unlabeled genes for a set of negatives.

## 2.3 Benchmark Development

The quality of a benchmark set is crucial in assessing a new prediction algorithm. We therefore developed a number of benchmarks designed to correspond as closely as possible to real situations in which prediction algorithms are used. Crucially, for all our benchmarks, we sought to minimize the overlap between the networks used for prediction and the dataset used for testing. This section will detail the three benchmarks we developed: one based on the GO, another on RNAi screens and a third on screens for long lived mutants in fission yeast. Table 2.3 summarises the benchmarks and how they were used.

### 2.3.1 GO Rollback Benchmark

This benchmark was built to mimic a CAFA style prediction competition: prediction was seeded with gene sets derived from GO annotations made prior to a specific cut-off date, using networks also pre-dating the cut-off. Performance was assessed based on how well the algorithm predicted new annotations made after the cut-off date. We call this a 'rollback' benchmark.

Specifically, predictions were seeded using sets of proteins labelled with the same GO term in 2009 and using functional association networks also from 2009. Performance was evaluated by considering new genes having acquired the label since 2009 as true positives. This GO rollback benchmark was constructed using data from yeast (*Saccharomyces cerevisiae*) and fly (*Drosophila melanogaster*). Evaluation sets were created from the Biological Process (BP) branch of the GO tree, using terms of level 5 and above in the GO tree (i.e. level 5 and

more specific). For each GO term, proteins annotated to the term prior to 2010 were taken as the seed set and those having acquired the label later as the test set. GO annotations were filtered by evidence code in order to 1) ensure high quality seed and test sets and 2) avoid predicted annotations, thus minimizing dependence between network data and test set. Specifically, only annotations derived from author or curator statements and directly from experiment were used (corresponding to evidence codes: IC, IDA, IMP, TAS and NAS).

Proteins not present in any of the functional association networks were ignored and categories with no proteins in the seed or novel set were excluded. This resulted in sets of 760 (yeast) and 967 (fly) GO terms.

A potential disadvantage with this benchmark is that it relies on the GO to provide training and testing examples. As discussed in the introduction, the use of GO in assessing prediction algorithms has been criticised for a number of reasons, including biases in how GO annotations accumulate and the inability to distinguish between true negatives and unlabelled genes. While the GO rollback benchmark will allow us to explicitly explore these effects, additional benchmarks were constructed to provide a complementary perspective on predictive performance.

### 2.3.2 Phenotypic RNAi Benchmark

The purpose of the phenotypic RNAi benchmark was to further reduce the overlap between the functional association networks and testing dataset. In the GO rollback benchmark, the labels used for testing are not necessarily independent of the networks - for example, if functional association networks are, directly or indirectly, driving label acquisition.

To avoid this potential issue, we designed an additional benchmark using knock-out screen: the functionally associated genes sets are composed of genes giving rise to the same phenotype when knocked out. The data was downloaded from the GenomeRNAi database [209], a repository containing phenotypes from RNAi screens in human and fly. Each gene set corresponds to a screen for a particular phenotype - lists of phenotypes and the number of genes giving rise to this phenotype when knocked out are given in Table 2.1 (for fly) and Table 2.2 (for human). To ensure independence from the network data, only genome-wide screens (as opposed to targeted screens, which tested only a subset of the genome) performed after 2009 were considered.

Similarly to the GO benchmark, the networks used for prediction pre-dated the test set. Thus, because the knock-out screens used were genome wide and therefore independent of any prior knowledge and all other data pre-dated these screens, this benchmark ensures independence of the networks and gene sets. Thus, the problems associated with cross-validation on the GO benchmark are not relevant for the phenotypic benchmark. We therefore evaluated performance using cross-validation, using 5 folds as a compromise between precision and time

needed to perform the benchmarking.

| Screen | Assay used to detect phenotype | Genes |
|---|---|---|
| Muscle morphogenesis and function | Posture, locomotion, flight and viability | 7955 |
| Heat nociception | Noxious heat avoidance and viability | 8270 |
| Notch induced transcription | Notch pathway reporter | 320 |
| NF-kB pathway regulation | Toll and Imd pathway Drosomycin reporter | 16 |
| Akt-TOR pathway negative feedback regulation | dAkt phosphorylation | 111 |
| HIF dependent transcription | Hypoxia inducible HRE reporter | 399 |
| G2-M DNA damage checkpoint regulation | Histone H3 phosphorylation | 157 |
| Self-renewal and differentiation in neural stem cells | Number and size of neuroblasts, ganglion mother cells, intracellular GFP aggregates and viability | 524 |
| Notch pathway regulation | Notch pathway reporter | 743 |
| Adiposity regulation | Total fly triglyceride expression | 7330 |
| S2 cell spreading | alpha-tubulin and actin protein expression | 217 |
| Secretory pathway regulation | BiP signal peptide and firefly luciferase fusion protein expression | 239 |
| RTK-Ras-ERK pathway regulation (in S2R+ cells) | ERK phosphorylation | 2021 |
| RTK-Ras-ERK pathway regulation (in Kc167 cells) | ERK phosphorylation | 2049 |
| Wg pathway regulation | WgRluc and sFluc protein expression | 304 |
| Hippo pathway regulation | Hippo pathway reporter | 9276 |
| Srp/Lz-induced transcriptional activation | Srp/Lz-induced transcription reporter (PO45) | 113 |
| Immune deficiency pathway regulation | Immune deficiency pathway reporter | 25 |

Table 2.1: List of the gene sets making up the phenotypic benchmark in fly. Each gene set corresponds to one genome-wide RNAi screen. The genes in the gene set give rise to the same phenotype when knocked-out. The table lists the phenotype assayed in each screen and the number of genes associated with each phenotype. The data was downloaded from the GenomeRNAi database [209].

### 2.3.3 Fission Yeast Ageing Benchmark

As a further method of validation, the prediction algorithms were benchmarked on an experimentally derived set of novel (i.e previous undiscovered) long-lived mutants in fission yeast (*Schizosaccharomyces pombe*) (see [214] for details). Predictions were seeded using known long-lived mutants *clg1*, *pef1* [29], *pma1* [97], *sck2* and *pka1* [202].

| Screen | Assay used to detect phenotype | Genes |
|---|---|---|
| Homologous recombination DNA double-strand break repair (HR-DSBR) | (HR-DSBR) DR-GFP reporter | 265 |
| HeLa cell morphology | Cell morphology | 609 |
| Self-renewal and pluripotency in human embryonic stem cells | POU5F1 protein expression | 384 |
| Vaccinia virus (VACV) infection | Number of influenza A H1N1 (A/WSN/33) infected cells and viral polymerase protein expression | 222 |
| TP53 interactions | TP53 protein expression and viability | 651 |
| Human papillomavirus oncogene expression regulation | HPV18 LCR reporter activity | 362 |
| Combinatorial effect with c-Myc | Viability (synthetic lethal) | 292 |
| Centrosome clustering | alpha-tubulin protein expression | 64 |
| DNA damage regulation after ionizing radiation | Ionizing radiation sensitivity | 286 |
| Proliferation and survival of human cancer cell lines | Viability | 477 |
| Homologous recombination DNA double-strand break repair (HR-DSBR) | (HR-DSBR) DR-GFP reporter and DNA content | 171 |
| TRAIL-induced apoptosis (1) | Viability | 178 |
| TRAIL-induced apoptosis (2) | Viability (synthetic lethal) | 25 |
| Negative-strand RNA virus infection (1) - vesicular stomatitis virus (VSV) | VSV-eGFP protein expression and DNA content | 41 |
| Selective autophagy regulation | Sindbis virus (SIN) capsid SIN-mCherry.capsid and autophagosome GFP–LC3 protein expression | 29 |
| Combinatorial effect with neratinib | Viability (synthetic lethal) | 10 |
| Regulation of FOXO1 nuclear localization | EGFP-FOXO1a protein expression and DNA content | 99 |
| Vaccinia virus (VACV) infection | Vaccinia virus VACV IHD-J/GFP protein expression and DNA content | 1978 |
| Combinatorial effect with MLN4924, a NAE inhibitor | Viability (synthetic lethal) | 187 |
| Non-small cell lung cancer (NSCLC) cytotoxicity (1) | shRNA abundance | 246 |
| Oncolytic Maraba rhabdovirus infection (1) | Viability | 122 |
| hepcidin regulation | hepcidin::fluc mRNA expression | 286 |
| Negative genetic interaction with BLM | shRNA abundance | 136 |
| Negative genetic interaction with MUS81 | shRNA abundance | 112 |
| Negative genetic interactions with PTEN | shRNA abundance | 107 |
| Negative genetic interaction with PTTG1 | shRNA abundance | 98 |
| Negative genetic interactions with KRAS | shRNA abundance | 197 |

Table 2.2: List of the gene sets making up the phenotypic benchmark in human. Each gene set corresponds to one genome-wide RNAi screen. The genes in the gene set give rise to the same phenotype when knocked-out. The table lists the phenotype assayed in each screen and the number of genes associated with each phenotype. The data was downloaded from the GenomeRNAi database [209].

| Benchmark Name | Organism(s) | Purpose | Performance Evaluation |
|---|---|---|---|
| GO Rollback | Yeast, Fly | Selecting PLS parameters | Cross-validation on annotations known prior to the cut-off date |
| GO Rollback | Yeast, Fly | Comparing Compass and GeneMANIA | How well annotations made after the date-cut off were predicted |
| RNAi Phenotypic | Fly, Human | Comparing Compass and GeneMANIA | Cross-validation |
| Ageing | Fission yeast | Comparing Compass and GeneMANIA | Prediction was seeded with long-lived mutants known prior to the screen and evaluated based on how well the mutants identified in the screen were predicted |

Table 2.3: Summary of the different benchmarks used in this chapter. The GO Rollback benchmark mimics CAFA style prediction competitions by using training labels and networks from prior to a specific cut-off date and evaluating the performance on annotations made after this date. This benchmark was also used in selecting the PLS parameters. In the RNAi phenotypic benchmark, functionally associated gene sets are derived from genes giving rise to the same phenotype in an RNAi knock-out experiment. The ageing benchmark uses an experimentally identified set of long-lived fission yeast mutants not previously known in the literature. Thus, in both the RNAi and ageing benchmarks, the gene sets used for testing are derived from genome wide screens and the networks used for prediction pre-date these screens: this benchmark is therefore free from the problems of information transfer associated with the CAFA style GO benchmark.

## 2.4 Preliminary Work

### 2.4.1 Regression and Support Vector Machines

We initially compared a number of approaches on the yeast GO benchmark. Building on the work by Heriche et al. [82], who used a commute time kernel combined with a nearest neighbour approach (see Section 2.2), we investigated the effect of combining a commute time kernel with a regression based approach and with a support vector machine classifier. Performance was measured as the area under a receiver operating characteristic (ROC) curve.

The relative performance of these methods is shown in Figure 2.3. All methods appear to perform almost identically (average error rate of about 0.3, corresponding to an AUC of about 0.7). While small difference in AUC may actually translate to relevant differences from a practitioner's point of view, neither of the methods seemed promising enough to warrant further exploration.

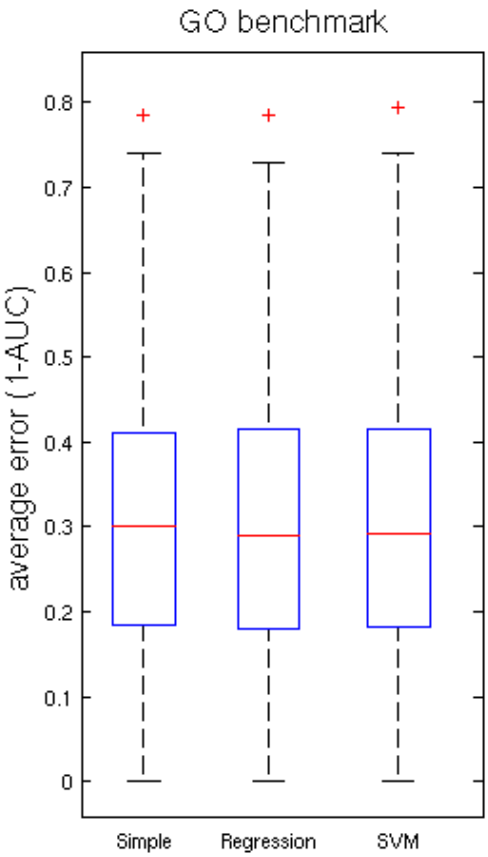


Figure 2.3: The relative performance of different prediction approaches on the GO benchmark. **Simple** is the nearest neighbour approach using the commute time kernel employed in the work by Heriche et al. **Regression** is a linear regression performed on the commute time kernel. **SVM** is prediction using a support vector machine and the commute time kernel. The figures show the prediction error, measured as 1 - AUC (area under ROC curve).

### 2.4.2 Dimensionality Reduction Approaches

Next, we explored dimensionality reduction approaches: these methods project the data into a lower dimensional space, in which a regression is then performed. In general terms, these approaches have the advantage of potentially reducing noise by removing non-informative dimensions. However, selecting the right number of dimensions to use in the regression is difficult. We cannot simply choose the number of dimension giving optimal performance in our prediction task (i.e. predicting novel labels): this uses information (the novel labels) we would not have access to in a genuine prediction context. Instead, the number of dimensions needs to be chosen based on information available at the time of prediction: performance on the seed set.

We therefore compared the performance of a principle component analysis (PCA) approach and a partial least squares (PLS) regression on both the seed set (assessed by cross validation) and on the novel genes (assessed by how highly the novel genes were ranked).

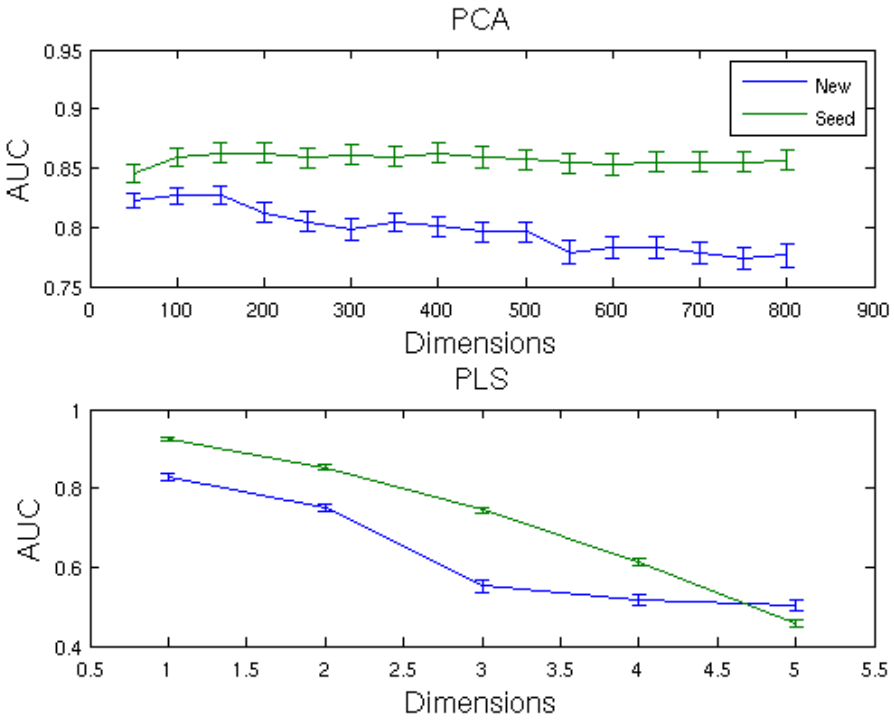


Figure 2.4: The performance of dimensionality reduction approaches (PCA and PLS) on the yeast GO benchmark. The figure shows performance (measured by AUC) as a function of the number of dimensions included, for both the novel labels and on the labels known at the cut-off date. The performance on the seed set was estimated by two-fold cross-validation. This performance is displayed because cross-validation on the seed set is necessary to select the number of dimensions to use. Error bars represent standard error of the mean.

As shown in Figure 2.4 (compared to Figure 2.3), both these methods outperformed our previous approaches (with AUC $> 0.8$ for PLS and PCA, compared to AUC of about 0.7 for our previous methods). However, the two algorithms differ in their robustness to the choice of dimensions. For PCA, the performance on the seed set is relatively robust across a range dimensions, with 400 dimensions giving optimal performance. On the novel set, on the other hand, optimal performance is achieved between 100 and 150 dimensions and deteriorates with the addition of further dimensions. This difference in behaviour is consistent with our hypothesis about the existence of overlap between the seed set and the functional association networks: there is information within the functional association networks which is helpful in characterizing the seed set, but is not useful in predicting new labels.

With PLS, the number of optimal dimensions between seed set and novel set is consistent: performance is maximized using a single dimension. This is in line with previous work recommending the use of K-1 dimensions for PLS discriminant analysis, where K is the number of classes [129].

Thus, while, at optimal parameter values, PCA would outperform PLS (see

Figure 2.4), the parameter values chosen by cross-validation lead to PLS outperforming PCA (0.832 vs 0.801). For this reason, we focused on PLS for further development and testing.

We also looked at PLS parametrization on the fly GO benchmark. Consistent with the results in yeast, the seed set gave optimal performance using 1 dimension (see Figure 2.5). For the novel proteins, performance is very slightly improved with the addition of an extra dimension.

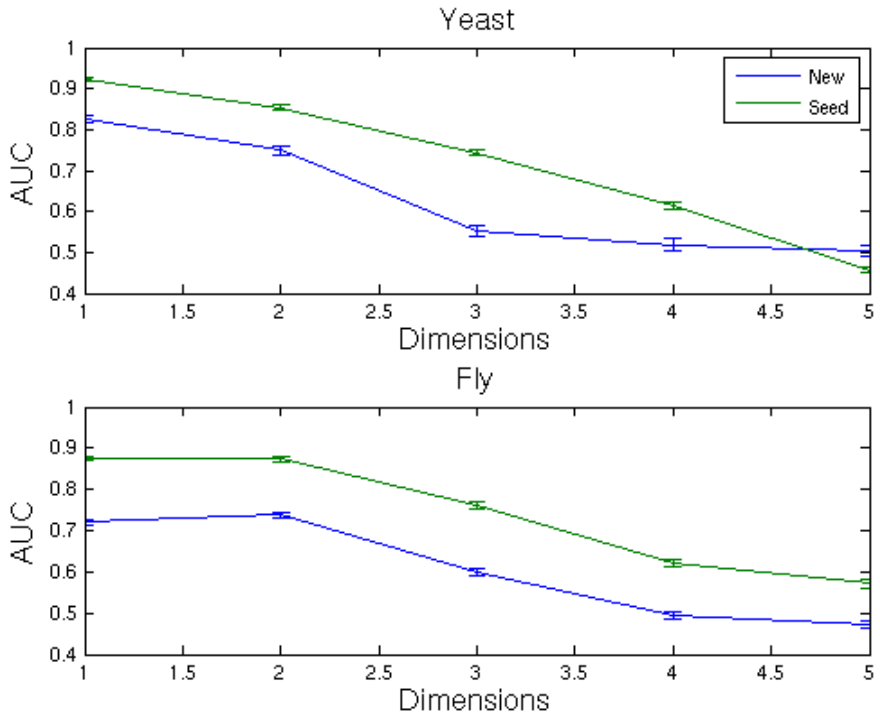


Figure 2.5: Compass performance on the GO yeast and fly benchmark sets, using different number of dimensions for the PLS regression. Performance is measured by area under ROC curve (AUC). Performance is shown estimated from cross-validation on the seed set ('seed') and prediction of new labels ('new'). Error bars represent standard error of the mean.

We also explored setting parameters separately for each GO term, but found this was detrimental to performance. Therefore, we used average performance across all GO terms to select a parameter value to use for all terms.

## 2.5 Algorithms

### 2.5.1 Compass Algorithm

We developed an algorithm based on a **com**mute-time kernel on the functional association network, followed by a kernelized version of a **pa**rtial least **s**quare**s** (PLS) regression (Compass). The Compass algorithm takes the following steps:

1. The commute-time kernel $K_{CT}$ [250] of the functional association network

is computed as laid out in Section 2.2. The commute-time kernel assumes the network has one connected component (i.e. all nodes are reachable from all nodes). In this work, if functional networks had more than one connected component, only the largest component was considered, as this resulted in the elimination of a very small minority of the nodes. For networks with larger or more numerous smaller components, each component can be treated separately.

2. The kernel matrix is normalized:

$$K_{CT}^{norm}(i,j) = K_{CT}(i,j)/\sqrt{K_{CT}(i,i) * K_{CT}(j,j)}$$

This step is taken to decrease the effect of nodes with large degree.

3. All genes not part of the seed set are treated as negative examples. The reason for this choice was two-fold: firstly, more sophisticated methods of negative example selection are GO-specific, whereas we seek to develop a general purpose tool. Furthermore, as discussed earlier, the inclusion of negative examples can have a detrimental effect on performance [251]. Thus, if $n_+$ is the set of known positives, then the target vector $y$ for the PLS regression is given by $y(i) = 1, \text{if } i \in n_+, \text{else } y(i) = -|n_+|/|n|$, where $n$ is the total number of genes so that the elements of $y$ sum to zero.

4. We perform the PLS regression using the implementation by Shawe-Taylor and Cristianini [213], using a single dimension in the regression. The predicted variable, $\hat{y}$, gives scores for all non-labelled genes reflecting their likelihood of belonging to $n_+$.

**Example of Compass Usage**

Figure 2.6 summarises the inputs and outputs of the Compass algorithm: the inputs are a list of genes of interest and functional association data and the output is a list of genes, ranked by likelihood of functional association with the genes of interest.

To illustrate how the algorithm is used, Table 2.4 gives the results of prediction using the fission yeast ageing benchmark. The algorithm is given the known long-lived mutants *clg1*, *pef1*, *pma1*, *sck2* and *pka1*, and the STRING functional association network for fission yeast. The Table shows the top 20 candidate genes returned by the Compass algorithm. Many of these 20 genes are associated with the cellular stress response and the list is enriched for GO categories relating to the cellular response to nutrients. There is a well documented link between stress, nutrient status and ageing [62], suggesting some of these putative longevity-related genes may indeed be worth further exploration.

| Gene | Description | Additional notes |
|---|---|---|
| SPBC1347.11 | stress responsive orphan 1 | Involved in oxidative stress response |
| SPCC4B3.07 | negative regulator of Ofd1 | Involved in cellular response to hypoxia |
| SPBC3D6.02 | But2 family protein | |
| SPBC354.07c | oxysterol binding protein (predicted) | |
| SPCC1494.08c | conserved fungal protein | |
| SPAC19E9.03 | cyclin Pas1 | Involved in regulation of cell cycle and transmembrane transport |
| SPAC24B11.06c | MAP kinase Sty1 | Involved in stress-activated MAPK cascade |
| SPAC8C9.03 | cAMP-dependent protein kinase regulatory subunit Cgs1 | Involved in nucleocytoplasmic transport and the stress response |
| SPAC11G7.01 | serine-rich Schizosaccharomyces specific protein | |
| SPAC13G6.10c | cell wall protein Asl1, predicted O-glucosyl hydrolase | involved in carbohydrate metabolic processes |
| SPBC1271.12 | oxysterol binding protein (predicted) | |
| SPBC16E9.13 | serine/threonine protein kinase Ksp1 (predicted) | Involved in cell signalling |
| SPAC26F1.10c | tyrosine phosphatase Pyp1 | Involved in MAPK signalling |
| SPCC1753.02c | G-protein coupled receptor Git3 | Involved in glucose mediated signalling pathway |
| SPBC19C7.03 | adenylate cyclase | Involved in glucose mediated signalling pathway |
| SPBC713.11c | plasma membrane proteolipid Pmp3 | |
| SPBC336.12c | MBF transcription factor complex subunit Cdc10 | |
| SPBC3E7.15c | sphingosine N-acyltransferase Lac1 | |
| SPAC1399.03 | uracil permease | |
| SPAC31G5.11 | cAMP-independent regulatory protein Pac2 | |

Table 2.4: The table gives the fist twenty genes returned by the Compass algorithm in response to a query list of genes with long-lived knock-out phenotypes. The list is enriched for GO categories relating to signalling and cellular glucose response.

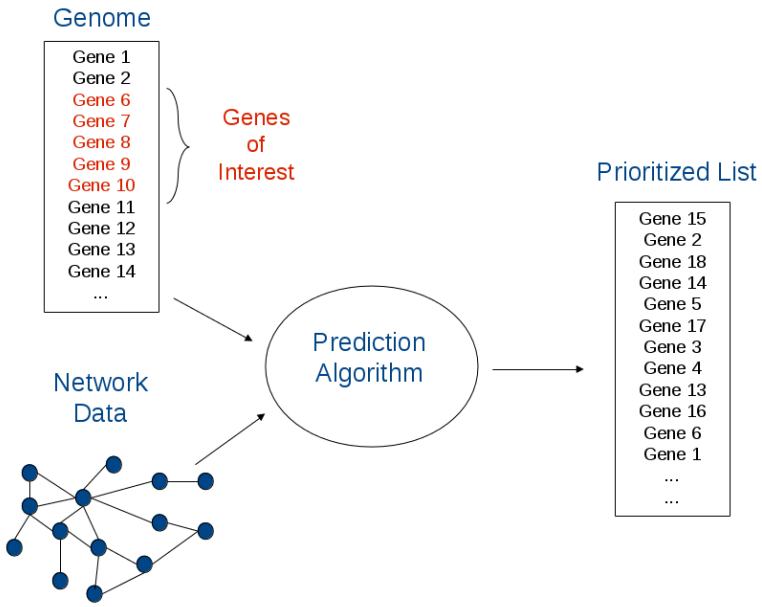


Figure 2.6: Summary of Compass' inputs and outputs: the algorithm is given a list of genes of interest and functional association data in the form of a network. The algorithm estimates how likely genes are to be functionally associated with the gene set of interest and returns a list of candidate genes ranked according to this score.

### 2.5.2 GeneMANIA algorithm

The performance of the Compass algorithm was compared to the algorithm used by the GeneMANIA [244] web-server. This section will discuss the GeneMANIA algorithm in detail. Briefly, GeneMANIA follows the following steps:

1. Network rescaling: the weights of all edges in the functional association networks are re-scaled to reduce the impact of high degree nodes.

2. Network scoring: All networks are given a score reflecting their relevance to the query set.

3. Network combining: Networks are combined, weighting each network according to the relevance score.

4. Prediction: Genes are scored according to likelihood of belonging to the pathway using a label propagation algorithm on the combined network.

### Network Scaling

$A$ is the adjacency matrix of the functional association network. $\hat{A}$, the re-scaled adjacency matrix is given by

$$\hat{A}(i,j) = A(i,j)/\sqrt{(\sum_i A(i,j) * \sum_j A(i,j))}$$

### Weighting

To compute a linear combination of the functional association networks, $W = \sum_d \alpha_d \hat{A}_d$, GeneMANIA calculates the weights $\alpha_d$ the following way:

An 'ideal' vectorized network, $t$ is constructed:

$$t(i,j) = \begin{cases} (n^-)^2/(n^+ + n^-) & \text{both i and j in pathway} \\ -(n^- n^+)/(n^+ + n^-) & \text{one of i or j in pathway} \\ 0 & \text{i and j both out of pathway} \end{cases}$$

where $n^+$ is the number of genes in the pathway and $n^-$ is the number of non-pathway genes connected to the pathway genes.

GeneMANIA then chooses $\alpha$ to minimize $(t - X\alpha)^T(t - X\alpha)$, where $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_d]^T$ and $X$ is a matrix with d columns, each containing a vectorized form of the functional networks (where non-pathway, non-pathway edges are excluded). The optimization is constrained by $\alpha_1, \alpha_2 \ldots >= 0$.

### Prediction

The target vector, $y$, is computed as $y \in +1, k, -1$ if gene $i$ is positive, unlabeled or negative receptively and $k = \frac{n^+ - n^-}{n}$.

Once the functional networks are combined, non-pathway genes are ranked according to likelihood of belonging to the pathway by solving the following optimisation problem:

$$\hat{\mathbf{y}} = \arg\max_{\hat{y}} \sum_i (\hat{y}_i - y_i)^2 + \sum_i \sum_j W_{ij}(\hat{y}_i) - \hat{y}_j)^2$$

which can be re-written in matrix notation as:

$$\mathbf{f} = \arg\max_{\hat{y}} (\hat{\mathbf{y}} - \mathbf{y})'(\hat{\mathbf{y}} - \mathbf{y}) + \hat{\mathbf{y}}'\mathbf{L}\hat{\mathbf{y}}$$

where $\mathbf{L}$ is the Laplacian matrix. The solution to this equation can be found by solving $\mathbf{y} = (\mathbf{I} - \mathbf{L})\hat{\mathbf{y}}$, that is: $\hat{\mathbf{y}} = (\mathbf{I} - \mathbf{L})^{-\mathbf{1}}\mathbf{y}$.

It is worth noting the similarity between GeneMANIA's label propagation algorithm and the random forest kernel $K_{RF} = (I + L)^{-1}$. Using a nearest neighbour approach with the random forest kernel will therefore give:

$$\hat{\mathbf{y}} = (\mathbf{I} + \mathbf{L})^{-\mathbf{1}}\mathbf{y}$$

### 2.5.3 Network Construction and Weighting

Because the prediction approaches explored in this chapter can be applied to any functional association network, network construction itself is not the focus of this chapter. However, as functional association network construction was necessary for benchmarking, we will briefly outline the approach taken in building these networks.

Functional association networks were downloaded from STRING database (version 8.1) [101]. STRING collates information about different indicators of functional association (conserved genome neighborhood, gene fusion, phylogenic co-occurrence, co-expression, database imports, large-scale experiments and literature co-occurrence) and weights interactions based on how well these interactions correspond to shared membership in KEGG pathways.

Different networks were combined into a single network simply by adding the adjacency matrices. Because associations in the individual networks are weighted on the same scale, this automatically introduces a weighting of the information sources. Unlike GeneMANIA's weighting algorithm, however, this weighting is not specific to the seed set. STRING also provides a pre-computed combined network, based on a naive Bayes approach. However, GeneMANIA's seed-specific weighting of different networks requires distinct networks - therefore, to achieve a fair comparison between GeneMANIA and Compass, we elected not to use this pre-computed network.

## 2.6 Comparison to GeneMANIA

### 2.6.1 Results Summary

Table 2.5 summarises the results of comparing Compass to GeneMANIA on a number of benchmarks.

| Benchmark set | Compass | GeneMANIA |
|---|---|---|
| RNAi (Fly) | 0.681 | 0.674 |
| RNAi (Human) | 0.636 | 0.625 |
| RNAi (Combined) | 0.654 | 0.644 |
| Ageing (Fission Yeast) | 0.713 | 0.613 |
| GO Yeast (all) | 0.832 | 0.803 |
| GO Fly (all) | 0.722 | 0.738 |

Table 2.5: Performance of Compass and GeneMANIA on the RNAi, Ageing and GO benchmarks as measured by AUC. In the RNAi benchmark, the human data consisted of 27 sets of functionally related genes and the fly data of 18 sets. In the GO benchmark, the full yeast set consisted of 760 terms and the fly set of 967 terms. The fission yeast screen consisted of 15 mutants identified in a genome-wide screen.

### 2.6.2 RNAi Benchmark

In the RNAi benchmark, functionally related genes sets are derived from an RNAi interference screen: the sets are formed of genes which, when knocked out, yield the same phenotype. Figure 2.7 shows the comparative performance on the RNAi benchmark, as estimated by five fold cross validation. As summarised in Table 2.6 Compass significantly outperforms GeneMANIA on this benchmark. When considering the fly benchmark alone, although Compass outperforms GeneMANIA, the difference between the two algorithms is not significant. It should be noted that the fly benchmark set is relatively small (n = 18) - it is therefore possible that the fly benchmark does not give us sufficient power to detect a statistically significant difference between the algorithms.

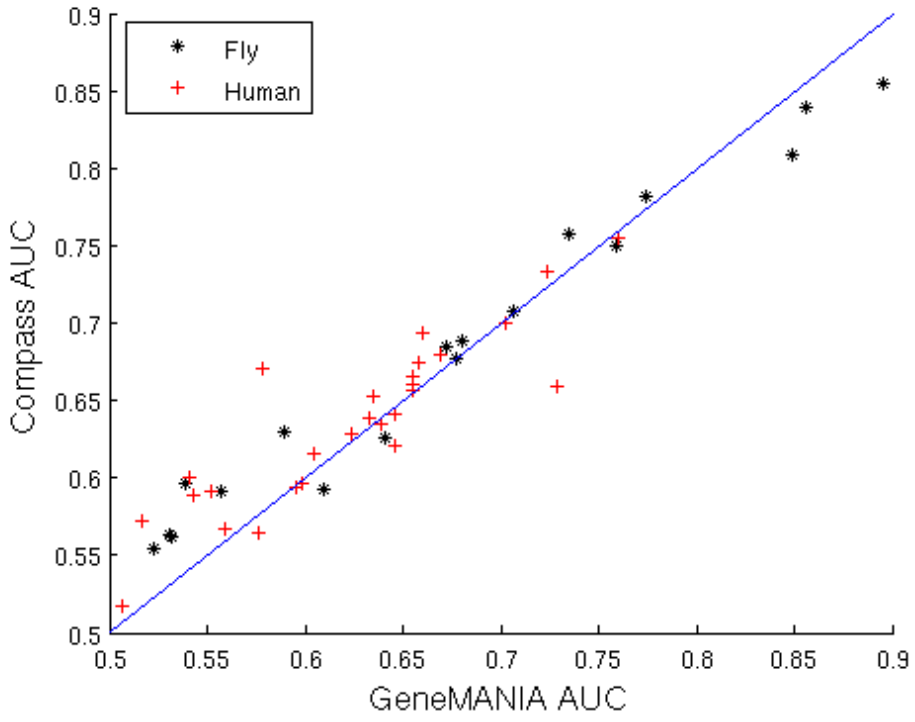


Figure 2.7: Comparison of the performance of Compass and GeneMANIA on the RNAi benchmark. Each data point represents AUC for a functionally related set of genes, as estimated by five fold cross-validation.

| Benchmark set | Compass | GeneMANIA | p-value |
|---|---|---|---|
| RNAi (Fly) | 0.6814 | 0.6737 | 0.3271 |
| RNAi (Human) | 0.6360 | 0.6245 | 0.0174 |
| RNAi (Combined) | 0.6542 | 0.6442 | 0.0175 |

Table 2.6: Performance on the RNAi benchmark, as estimated by five fold cross-validation and measured by AUC. P-values are derived from a two-tailed Wilcoxon ranked sum test. The human data consisted of 27 sets of functionally related genes and the fly data of 18 sets

### 2.6.3 GO Benchmark

The relative performance of Compass and GeneMANIA at predicting novel GO labels was assessed using a roll-back benchmark (see Section 2.3). Average performance across all GO categories, as measured by AUC, was 0.832 with Compass and 0.803 with GeneMANIA for the yeast benchmark and 0.722 (Compass) and 0.738 (GeneMANIA) on fly.

Assessing the statistical significance of these results is not entirely trivial: because of the hierarchical structure of the GO, there can be considerable overlap in the sets of proteins annotated with different GO terms. For example, all proteins annotated as participating in the process *cation transmembrane transporter activity* will also be annotated with the term *ion transmembrane transporter activity*. Thus, the performance of a prediction algorithm on one of these GO sets is not independent of its performance on the other. This dependence between GO terms introduces a difficulty for statistical testing: statistical tests tend to assume independence of data points.

This problem has not been discussed particularly widely in the literature and, to our knowledge, no standard solution exists. Some authors choose to consider only the most specific level of the GO hierarchy, thus avoiding dependencies between parent and child terms. However, this does not guarantee independence: GO terms at the same level of the hierarchy may be closely related and thus be applicable to many of the same proteins. Furthermore, considering only the most specific level may introduce a systematic bias into the assessment process: the performance of a prediction algorithm may be dependent on the specificity of the predicted function.

To address this problem, we sought to generate a set of independent GO terms. Overlap between GO terms was here defined as $max(|n \cap m|/|n|, |n \cap m|/|m|)$ where $|m|$ and $|n|$ are the sets of proteins labelled with GO terms $m$ and $n$ respectively. GO terms with % overlap below a specific threshold were considered independent. To build the largest possible set of independent GO terms, the GO terms with overlap exceeding the threshold with the largest number of other GO terms was eliminated. This process was repeated until only independent GO terms remained.

There is a trade-off between the overlap threshold and the number of GO terms available for testing (and thus the power of the statistical test). Several thresholds were therefore explored (see Table 2.7) and results are consistent across thresholds: on the yeast benchmark, Compass significantly outperforms GeneMANIA, while on the fly benchmark GeneMANIA outperforms Compass.

### 2.6.4 Ageing Benchmark

The final benchmark was based on long lived mutants identified in a genome wide screen [214]. Prediction was seeded with long lived mutants known prior to the screen. On this benchmark, Compass outperformed GeneMANIA (0.7131

| Benchmark set | Compass | GeneMANIA | p-value (if applicable) |
|---|---|---|---|
| GO Yeast (all) | 0.832 | 0.803 | - |
| GO Yeast ($< 75\%$ overlap) | 0.831 | 0.789 | $4.67 * 10^{-4}$ |
| GO Yeast ($< 50\%$ overlap) | 0.837 | 0.816 | 0.0049 |
| GO Yeast ($< 25\%$ overlap) | 0.838 | 0.816 | 0.0127 |
| GO Fly (all) | 0.722 | 0.738 | - |
| GO Fly ($< 75\%$ overlap) | 0.709 | 0.726 | $1.42 * 10^{-4}$ |
| GO Fly ($< 50\%$ overlap) | 0.709 | 0.723 | 0.0014 |
| GO Fly ($< 25\%$ overlap) | 0.713 | 0.730 | 0.0033 |

Table 2.7: Comparison of Compass and GeneMANIA on GO benchmark sets with reduced overlap for statistical testing. The table shows the performance of the algorithms on a subset of the GO terms with $< 75\%$, $< 50\%$, $< 25\%$ overlap in labelled proteins. The number of GO terms in each category was 309, 207 and 130 for GO-terms for yeast, 440, 393 and 239 for fly).

vs 0.6129). The statistical significance of this result was evaluated by comparing how highly each long lived mutant was ranked by Compass and GeneMANIA, giving a p-value of 0.0168 (two-tailed Wilcoxon sing-rank test).

### 2.6.5 GeneMANIA weighting scheme

The Compass and GeneMANIA algorithms differ in two ways: firstly, the way prediction networks are combined (with GeneMANIA computing a query specific weighting) and in the prediction algorithm itself. Therefore, to ensure that the observed differences in performance were not simply due to GeneMANIA's seed-specific weighting of the networks, we also ran GeneMANIA without the seed-specific weighting step on the GO benchmark. This gave average performance of 0.800 and 0.747 for yeast and fly respectively (compared to 0.832 and 0.722 with Compass). The relative performance of the two algorithms was thus unchanged. Interestingly, for the fly dataset, GeneMANIA's performance is improved by removal of the seed-specific weighting.

## 2.7 Detailed Investigation of Prediction

In order to understand potential biases in our prediction algorithm or benchmarking paradigm, we studied prediction results in detail.

### 2.7.1 Cross-Validation vs Rollback

We compared prediction performance as evaluated by cross-validation on the seed set and using the rollback benchmark. As expected, performance was higher using cross-validation (see Figure 2.8), suggesting information transfer between the functional association network and the seed set causes the seed set to be 'too easy' to predict.

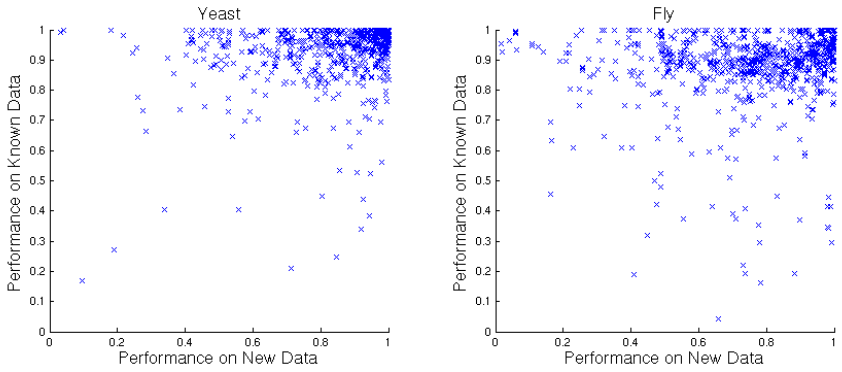


Figure 2.8: Comparison between Compass performance on new data and known data (as measured by two-fold cross validation) in yeast and fly benchmarks. Each data point represents performance on one GO term.

Furthermore, correlation between the two benchmarks was relatively low (Pearson's correlation coefficient of 0.260 for yeast and 0.073 for fly), indicating that cross-validation on known protein sets is not a particularly good indicator of performance when predicting novel proteins.

This low correlation could potentially interfere with parametrization because, in practice, parameter selection usually involves some form of cross-validation. Indeed, as discussed previously, cross-validation leads to a non-optimal choice of parameters for the fly benchmark.

### 2.7.2 Effect of Gene Degree on Label Predictability

Next, we sought to investigate the relationship between a gene's degree and how easily predictable new labels for that gene are: some authors have raised concerns that network-based prediction methods tend to favour high degree genes [64]. We investigated this in our benchmarks by taking the set of test genes and looking at the relationship between a gene's degree and its position in the prioritized list (thus, low ranking = high priority of being functionally associated with the seed set).

In the yeast GO benchmark, there is a significant negative correlation between degree and ranking for both Compass and GeneMANIA (SCC -0.4031 and -0.3266, respectively, $p < 10^{-60}$) (see Figure 2.9): genes with high degree tend to be easier to predict. Interestingly, the greatest difference in performance for the two methods is for low degree genes, where Compass clearly outperforms GeneMANIA.

In fly, on the other hand, the situation is very different: while GeneMANIA performs relatively consistently across degrees (slight negative correlation between ranking and degree, SCC -0.1009, $p < 10^{-15}$), Compass' performance is very dependent on degree (SCC -0.8189, $p < 10^{-60}$).

However, when the *same* network is used as a predictor for the RNAi phenotypic benchmark, both methods shows a similar dependence on degree (SCC

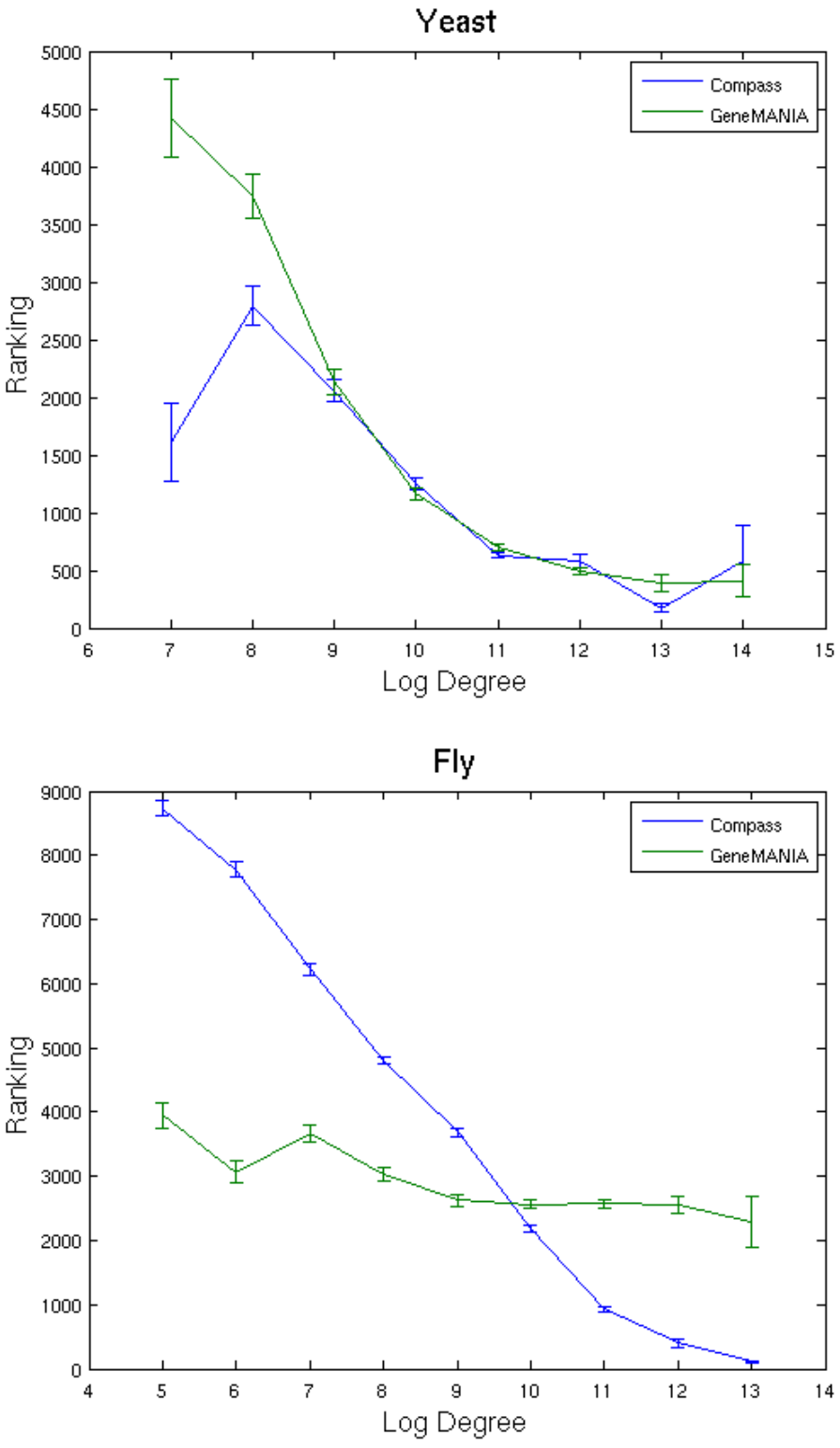


Figure 2.9: The effect of a gene's degree on its predictability in the GO Benchmark. The figure shows the average ranking of novel labels, grouped by the degree of the predicted gene. Error bars represent standard error of the mean.

-0.6117 and -0.6512 for Compass and GeneMANIA respectively), with Compass again outperforming GeneMANIA on low degree nodes (see Figure 2.9). This therefore suggests that the strong correlation seen between degree and perfor-

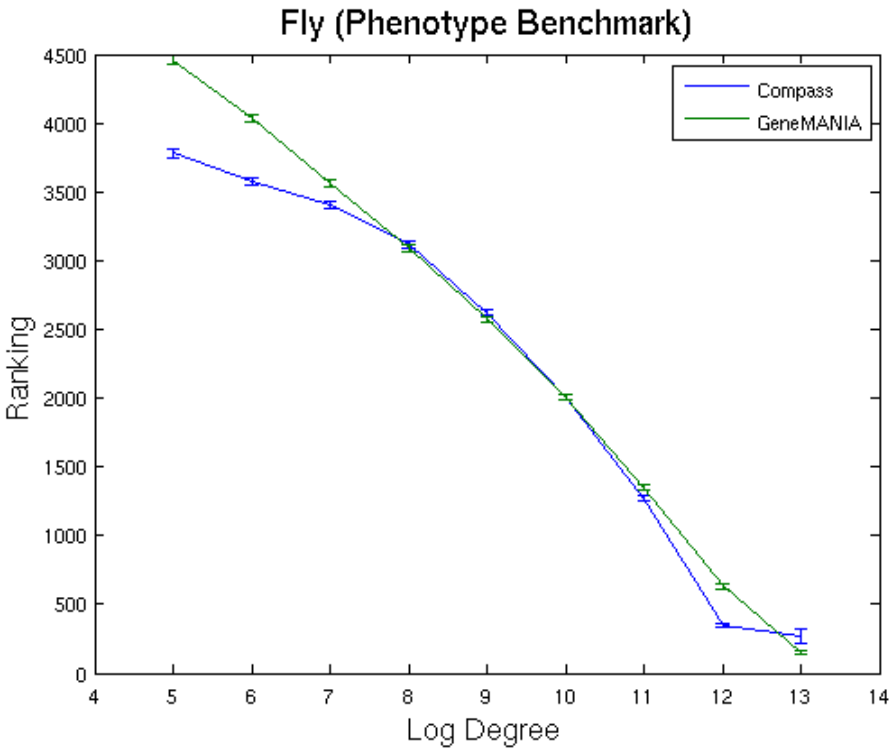


Figure 2.10: The effect of a gene's degree on its predictability in the fly phenotype benchmark. The figure shows the average ranking of novel labels, grouped by the degree of the predicted gene. The ease of prediction is measured as the ranking of the gene in a prioritized list (low ranking = easy to predict). Error bars represent standard error of the mean.

mance on the fly GO benchmark is not an inherent property of the network, but depends on the set of genes used to seed and evaluate the performance.

### 2.7.3 Effect of Discovery Date on Label Predictability

Next, using the GO rollback benchmark, we investigated the effect of a label's discovery date on how easy it was to predict. In yeast, later labels are more difficult to predict (see Figure 2.11): there is a significant positive correlation between date of discovery and ranking for both compass and GeneMANIA (SCC 0.2058 and 0.1629 respectively, $p < 10^{-15}$). In fly, a slight correlation is found for Compass, but not GeneMANIA (SCC 0.0267, $p = 0.033$).

### 2.7.4 Effect of Degree on Discovery of New Labels

Having identified two factors affecting how well Compass and GeneMANIA predict gene labels (the labelled gene's degree and the date at which the label was discovered), we were interested in whether there could be a potential interaction between these effects. We therefore looked at whether genes with high degrees were more likely to acquire new labels.

In yeast, overall, genes with high degrees tend to acquire new labels first (see

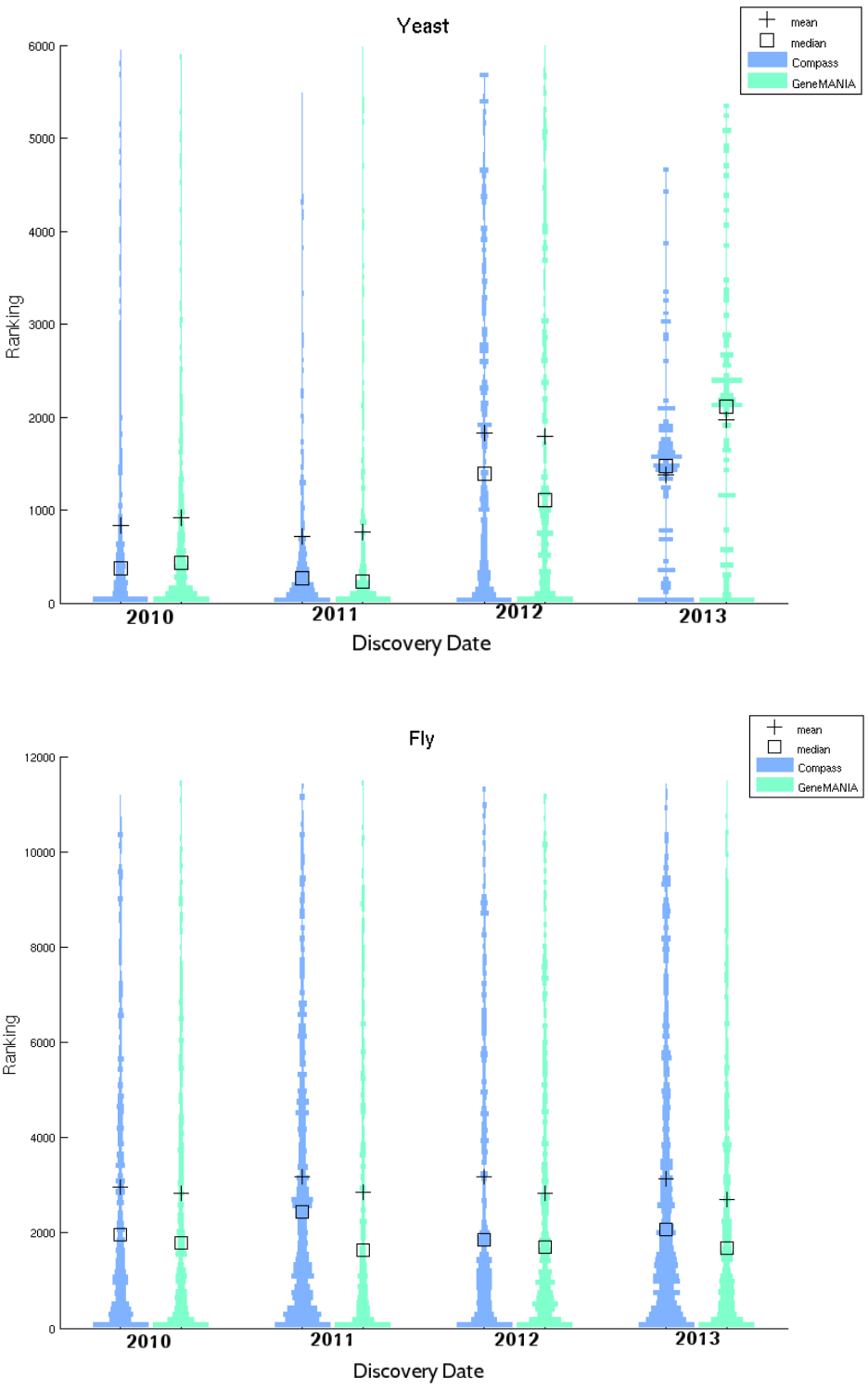


Figure 2.11: Relationship between date of how easy a label is to predict and the degree of the labelled gene. The ease of prediction is measured as the ranking of the gene in a prioritized list (low ranking = easy to predict). The width of each distribution has been normalized individually (i.e. only relative shape, not width, is relevant).

Figure 2.12): there is a significant negative correlation between degree and date of discovery (Spearman correlation coefficient (SCC) -0.1750, $p < 10^{-19}$). No

significant correlation was found in the fly data.

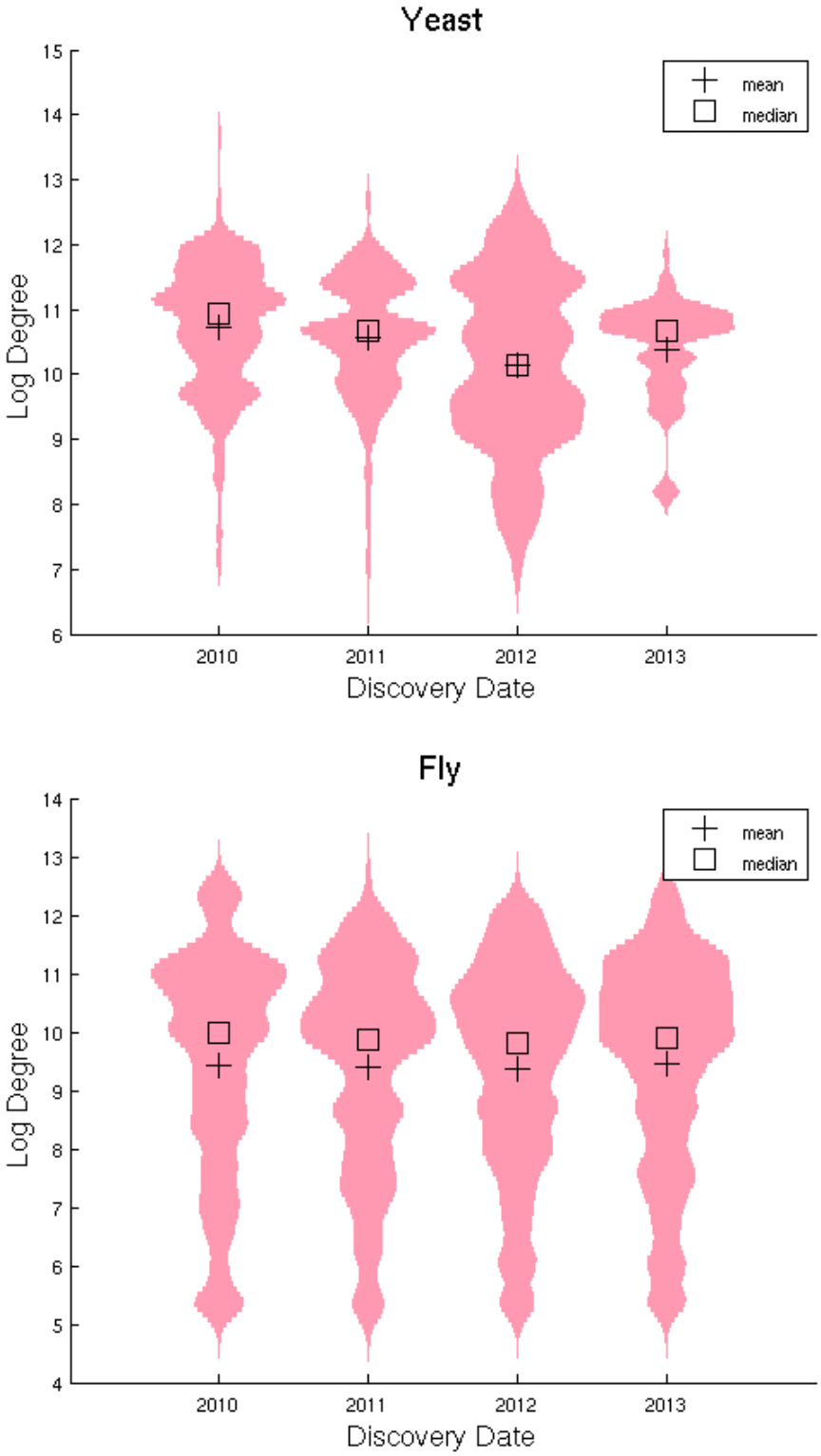


Figure 2.12: Relationship between date of discovery of a new label and the degree of the labelled gene on the GO rollback benchmark. The figure shows the distribution of degrees of the genes for which new labels were discovered during the course of each year. The width of each distribution has been normalized individually (i.e. only relative shape, not width, is relevant).

## 2.8 Discussion

### 2.8.1 Relative Performance of GeneMANIA and Compass

Overall, we find Compass outperforms GeneMANIA on a majority of our benchmarks. Compass performs better on the RNAi phenotype, yeast GO and fission yeast ageing benchmarks, while GeneMANIA outperforms Compass on the fly GO rollback benchmark.

Interestingly, while success at predicting a particular label is correlated with the degree of the labelled gene for both Compass and GeneMANIA on both the RNAi and GO benchmarks, this effect is clearly strongest for Compass on the fly GO benchmark, the only benchmark for which GeneMANIA outperforms Compass. This is not a feature of the fly network itself: on the RNAi benchmark, the degree-performance correlation of the two methods is comparable. The high degree dependency of Compass' performance on the fly GO benchmark therefore appears to be a result of the data used to seed and/or evaluate the performance. It is unclear what particular feature of the fly GO data results in this behaviour - further investigation of this effect could provide further insights into the factors determining the relative performance of the two algorithms.

Conversely, on the other benchmarks we studied, Compass outperforms GeneMANIA on nodes with low degree. These are also the benchmarks for which Compass outperformed GeneMANIA overall. This raises the possibility that a key feature of a successful predictor is the ability to successfully make predictions for genes for which less functional association information is available.

### 2.8.2 Further Investigation of the Effects of Network Quality on Predictive Performance

The reliability of the predictions made by Compass and GeneMANIA is dependent on the quality of the network data used by the algorithms. However, prediction algorithms may differ in how well they tolerate noisy network data and in their sensitivity to different types of noise (false edges versus missing edges, for example). Given the concerns over functional association network quality and completeness, understanding how an algorithm responds to noisy input data is valuable. It would be interesting to test Compass' dependence on network quality by introducing noise into the functional association networks by removing true edges and/or adding spurious edges to the network. Additionally, because String gives each interaction a score according to the reliability of the evidence for the interaction, further insight could be gained by experimenting with different reliability score thresholds for the input network. This could help understand whether the prediction algorithm performs better with a sparse high quality network or with denser but less reliable network data.

### 2.8.3 Cross-Validation and Parameter Selection

Our results also highlight potential difficulties when it comes to assessing how well prediction methods perform. Like previous studies [175], we find that cross-validation on known datasets overestimates the quality of our prediction methods: on the GO rollback benchmark when performance is measured on retrieval of new labels, results are consistently lower than when performance is measured by cross-validation on the labels known prior to the cut-off date. Furthermore, the correlation between these two measures is low, suggesting that performance on a cross-validation benchmark is not indicative of success in predicting new labels. Additionally, as seen from our results on the fly GO benchmark, cross-validation on the seed set may lead to non-optimal choice of parameters.

### 2.8.4 Temporal Effects

Our results on the yeast (but not fly) GO rollback benchmark also suggest an interesting temporal effect on the difficulty of predicting new labels: labels acquired a short time after the date of origin of the functional association networks (*'early new labels'*) are easier to predict than labels acquired several years later. This effect may be attributable to information transfer between databases. If the discovery of a new functional association clearly implicates a protein in a particular function, this function is 1) likely to be easily predictable from the functional association data and 2) become incorporated into GO rapidly and thus feature in the early new labels. In essence, this extends the problem of non-independent network and evaluation data from cross-validation to early new labels as well.

Even if the discovery of functional associations does not directly lead to functional annotation in GO, early new labels may still be non-independent if the discovery of new labels is guided by the state of functional association networks at the time. Indeed, a researcher interested in a particular pathway is likely to use the information available at the time to test for new pathway components. Thus, the network proximity of a hidden true positive to other genes in its GO category could be a major factor in both how soon the label is uncovered and how well the new label is predicted through guilt-by-association approaches.

It is difficult to untangle the temporal effect from the effect of gene degree on the predictive performance. Annotations for genes with high degree are easier to predict. In the GO benchmark, genes with high degrees tend to acquire annotations first. Therefore, the higher performance on the early new genes may be due to this degree effect, rather than a direct consequence of temporal effects. This would also explain why the correlation between date of discovery and ease of prediction is not seen in the fly data, where no correlation between degree and date of discovery was found.

These observations raise the question of why genes with a higher degree acquire labels sooner. Again, we hypothesis similar effects to those discussed

above: the discovery of new functional information is guided by what is currently known about function. Thus, we are more likely to discover more about the function of genes we are already familiar with. This is the 'rich get richer' effect discussed in the Introduction.

These temporal effects are not seen in the fly. The reasons for this are unclear. Investigating whether similar effects exist in other organisms would allow determining how general these temporal effects are and would potentially clarify why they are not seen in the fly data.

### 2.8.5 Problems with CAFA-Style Benchmarks

These observations raise the question of whether differences between early and late new labels affects what is being assessed by CAFA or MouseFunc style benchmarks. The time period between prediction and assessment (i.e. the time window allowed for new annotations to accumulate) is typically under a year - performance is therefore only assessed on the early new labels. If these labels are indeed biased by the processes discussed above, the ability to assess prediction methods' performance will also be affected. This problem is exasperated by the necessity of considering genes lacking a specific category labelled as negatives for that term - when they could actually represent hidden true positives [43]. This could lead to penalisation of methods ranking the 'more difficult' and not yet discovered labels higher than the more obvious ones. This leads to the concern that competition style benchmarks may encourage the building of tools to mimic experimental discovery as opposed to guiding it.

### 2.8.6 Conclusion and Further Work

We have proposed a novel guilt-by-association prediction algorithm for gene function prediction and compared its performance to GeneMANIA, a leading network based prediction algorithm. We find that the relative performance of the two algorithms is dependent on the benchmark set used, suggesting a one-size-fits all approach to function prediction may not be optimal. However, it is worth noting that on the benchmark sets where the functional association network was independent from the evaluation data (i.e. RNAi and ageing benchmarks), Compass consistently outperforms GeneMANIA.

We explicitly examine how the choice of a benchmark set affects perceived performance. We find that on the GO benchmark, performance assessed through cross-validation is not necessarily indicative of performance on new labels and that, for CAFA-style prediction challenges, the time lag between prediction and evaluation may significantly alter perceived performance. Thus, while these systematic evaluation efforts are crucial for meaningful comparison of prediction algorithms, there are questions over choice of benchmark sets and evaluation methods that remain to be addressed. It seems clear that simply relying on the GO for comparison of different algorithms is not sufficient.

Part of the difficulty in protein function prediction arises from the broadness of the concept of protein function. Indeed, it would be surprising if a single method performed optimally in all contexts. Thus, a potentially fruitful approach would be to focus on optimizing prediction algorithms for specific contexts, instead of seeking to build a general function predictor.

The work in the chapter has explored protein function prediction in a *function-centric* context: the algorithm is designed for users interested in finding novel proteins involved in a specific function. As mentioned previously, similar algorithms can also be applied to predicting the function of a specific gene. This is an important potential application for Compass-type methods: there is a growing gap between the number of sequenced genes and the number of functionally annotated genes. The usefulness of network-based prediction methods for this type of prediction is limited because there is often very little functional association data about unannotated genes. A strength of the Compass method is that the kernel used for prediction need not be derived from network-data. It would be straightforward to construct a kernel from structural similarity data, for example. Thus, if used with non-network input data, Compass could also be a valuable tool for predicting the function of unannotated genes.

# Chapter 3

# Identifying Genetic Interactions between Loss of Function Tolerant Genes

## 3.1 Introduction

One of the fundamental questions in modern biology is the relationship between genotype and phenotype. For a limited number of phenotypes (Mendelian diseases for example), this relationship is well understood. Overall however, how variation in genotype leads to variation in phenotype remains an open question.

Considerable efforts have been made to address this issue. Large scale projects mapping human genetic variation, such as the 1000 genomes project [1] which involves the complete sequencing of over 1000 human genomes, hold a lot of promise and provide rich data sets for computational approaches. One interesting observations arising from these efforts is the high frequency of non-functional genes in the genomes of healthy people. This tolerance to loss of function is surprising: redundant genes would be expected to be lost during the course of evolution. Therefore, this apparent redundancy (often referred to as loss of function tolerance) is of interest to researchers [143].

### 3.1.1 Loss of Function Variation

Loss of function (LoF) variants are mutations in protein coding genes (or, indeed, in functionally important non-coding regions) that lead to significant or complete loss of protein function. Traditionally, LoF variants have been assumed to be deleterious and therefore expected to occur only rarely. However, in light of recent whole genome sequencing studies, it appears that LoF variants may be more common than previously thought: estimates for the number of LoF variants carried by apparently healthy individuals rage from 100 to 800 [143].

However, a number of factors make discerning between apparent and genuine LoF tolerance challenging. Firstly, correctly identifying LoF variants is

problematic (see below), leading to a high probability of false calls. Secondly, if LoF variants occur only heterozygously, they may in fact be recessive disease mutations. Thirdly, even LoF variants appearing homozygously in healthy genomes may have a complex relationship to phenotype, such as being necessary only in specific genetic backgrounds or environmental conditions.

### 3.1.2 Challenges in LoF Variant Identification

The identification of LoF variants requires predicting whether changes in the coding sequence will result in a non-functional protein. Both the sequencing and prediction processes are associated with errors. It is therefore possible that high LoF variant frequency may be partially attributable to calling errors. Therefore, before studying the frequency of LoF variation in more detail, we must first address the question of false positives.

Potential sites for loss of function variants are particularly sensitive to sequencing errors [144]. Sequencing technologies involve DNA fragmentation, sequencing of these fragments (i.e. 'nucleotide calling') and mapping of the resulting reads onto a reference genome. Both base calling and mapping processes are error prone, with next generation, short-read sequencing technologies being particularly vulnerable to these mis-call and mis-mapping errors [151]. The reason this problem is particularly acute for potential loss of function sites is that variants disrupting protein function are under negative selection, which leads to lower variation in functional regions of the genome. Meanwhile, sequencing error is uniformly distributed across the genome. Thus, the signal (true variation) to noise (calling and mapping errors) is expected to be lower in functional regions than genome average, leading to a higher rate of false positive LoF variant calling [144].

Even discounting sequencing errors, the challenge of predicting which variants will give rise to non-functional protein remains. Generally, variants causing premature stops, shifts in the reading frame, splice site disruption or large scale deletion are categorised as LoF variants [144]. However, the true phenotypic consequence of these variants is not necessarily complete loss of function. For example, in some cases, a truncated transcript may produce a functional version of the protein. Furthermore, given the prevalence of alternative splicing in the human genome [108], we would expect to find some LoF variants affecting only a subset of a gene's transcripts.

Attempts have been made the mitigate these problems. MacArthur et al [143] filter candidate LoF variants from the pilot phase of the 1000 genomes project (185 sequenced genomes and 3000 putative LoF variants) based on sequence read mapping and quality, local sequence context, gene annotation and the predicted effect of nearby variants. This process filtered out the majority of the candidate LoF tolerant genes, giving the previously mentioned estimate of 100 high confidence LoF variants per genome. Thus, even using conservative

estimates, LoF tolerance is surprisingly common.

### 3.1.3 Interactions between Genes: Recessive LoF Variants and Epistasis

One potential explanation for the frequency of LoF tolerant genes is that some of these genes are only conditionally LoF tolerant: the gene is only necessary in specific genetic backgrounds or environmental conditions. Recessive disease alleles are the most straightforward example of this type of interaction: the disease phenotype only manifests in presence of both disease alleles. More complex interactions between genes at different loci are known as genetic interactions or epistasis and are considered to be fundamental in understanding complex disease [35]. Thus, differentiating between genuine LoF variants, recessive disease alleles and variants associated with complex disease would be of great interest from a clinical point of view.

### 3.1.4 Aims and Objectives

This chapter explores loss of function variation in the 1000 genomes project, identifying potential genetic interactions between a set of apparently loss of function tolerant genes - that is, genes for which variants are present homozygously in the genome of healthy individuals. First, we develop approaches for identifying pairs of potential genetic interactors and analyse a list of putative interactors. Secondly, we explore methods for identifying larger communities of potential genetic interactors and suggest a set of potential epistatic communities based on these methods.

## 3.2 LoF Data

The data used in this chapter was collected as part of phase 1 of the 1000 genomes project (1092 genomes) [1]. Prior to the work presented here, LoF variants were identified from the sequence data as outlined in [143]. Briefly, variations causing shifts in reading frame, splice overlap or premature stops within a coding region were classified as LoF variants (Figure 3.1). LoF variants occurring in only one allele were excluded - all remaining LoF variants (317 genes) thus correspond to homozygous loss of function.

This data can be presented as an occurrence matrix, $X$, of dimensions $m$ by $n$, where $m$ is the number of genes with LoF variants in the healthy genomes (317) and $n$ is the number of genome samples (1092). $X(i, j) = 1$ if sample $j$ has a homozygous loss of function variant of gene $i$ and 0 otherwise.

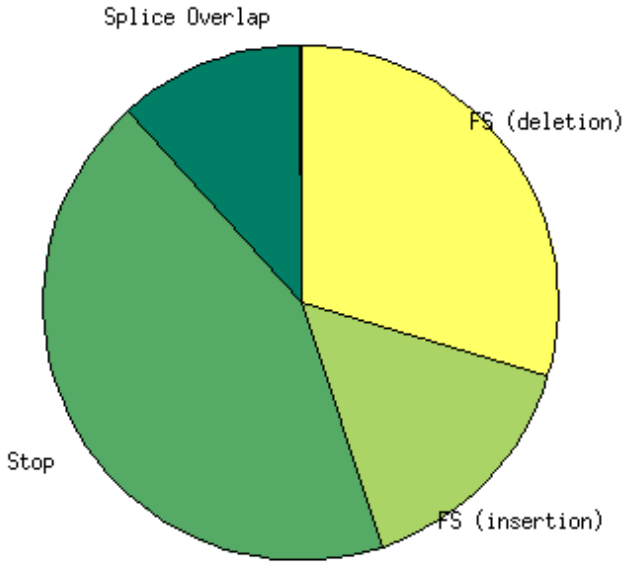


Figure 3.1: Types of variation predicted to lead to loss of function: frame shifts (FS), splice overlap and premature stops. The pie chart shows the proportion of the LoF variants in the 1000 genomes data set resulting from each type of variation.

## 3.3 Identifying Pairwise Genetic Interactions

The identification of genetic interactions in humans is challenging. In model organisms, particularly yeast, genetic interactions have been probed by inducing loss of function ('knocking out') two genes simultaneously and comparing the resulting phenotype to the single knock-out phenotypes. Double knock-outs can be performed in human cell lines, but this only allows identifying interactions that affect phenotype at the cell level. Epistasis in humans has also been studied using genome-wide association studies (GWAS) [135], using statistical methods to detect over-representation of single nucleotide polymorphism (SNP) pairs in individuals with a particular disease. This approach is potentially powerful, but also has drawbacks. Firstly, because individuals have a large number of SNPs - approximatively around 10 million per genome [124,195] - a large number of samples will be needed to confidently detect interactions between them. Secondly, detecting genetic interactions using GWAS requires focusing on interactions associated with a particular disease.

Here, we propose an alternative approach: we look for pairs of LoF variants that occur together in healthy genomes less often than expected (Figure 3.2). We are interested in genetic interactors that have detrimental effect on health:

thus, by definition, genomes with loss of function in both of these interactors would be less likely to appear in our sample of healthy genomes. LoF variants occurring together in healthy genomes less often than expected are therefore candidates for genetic interactors.

Our approach has the advantage of not having to focus on a specific disease. Additionally, because we are specifically interested in LoF variants instead of SNPs, we require fewer samples. The drawback, however, is that we are limited to the study of complete loss of function, and will therefore not be able to detect the consequences of more subtle variation. Additionally, while the sample size required is smaller than in GWAS, the 1000 genomes available to us may not provide enough statistical power to identify interactnewnightions between rarer LoF variants. Nevertheless, the accumulation of sequence data is on-going. Thus, even if interesting interactions are not found in the 1000 genomes data set, the methods developed will be applicable once more data is available.

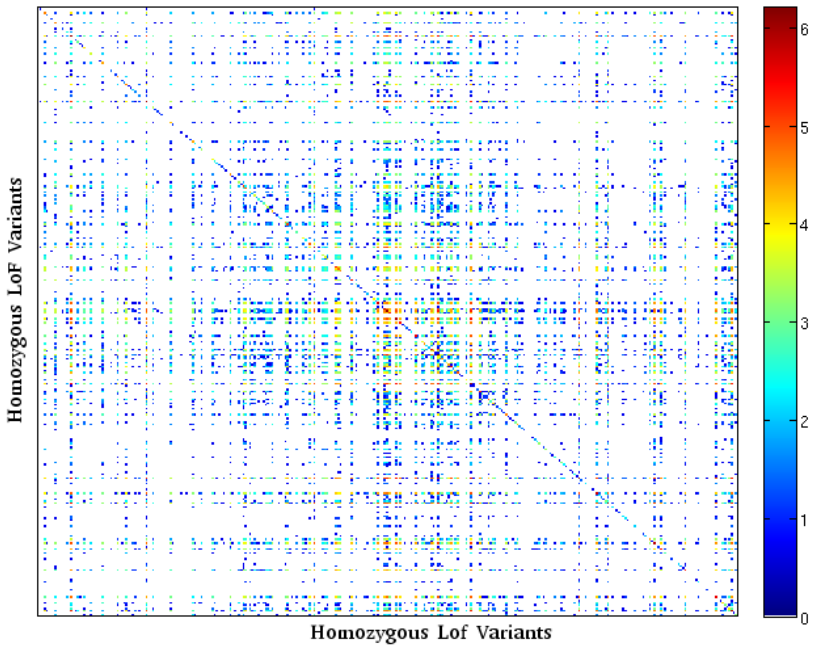


Figure 3.2: The Figure shows the LoF co-occurrence matrix: how often homozygous LoF variants at different loci appear in the same healthy genome in thousand genomes project data. The co-occurrence matrix, $A$, is computed as $A = XX^T$. The colour in the heatmap represents the number of samples in which both genes carry homozygous LoF mutantions (on a logarithmic scale with base $e$). We are interested in identifying gene pairs which co-occur less often than expected based on their prevalence in the genome data. Note that the diagonal entries in the matrix are all non-zero: they represent how many genomes each LoF variant occurs in. The diagonal appears discontinuous because of the figure resolution.

### 3.3.1 Hypergeometric Model

To identify pairs of LoF variants co-occurring at unexpectedly low frequencies, LoF co-occurrence was modelled using the hyper-geometric distribution. The probability $P(k)$ of LoF variants in gene A and gene B occurring in the same genome $k$ times is given by:

$$P(k) = \frac{\binom{a}{k}\binom{n-a}{b-k}}{\binom{n}{b}}$$

where $a$ is the number of samples LoF variant A occurs in, $b$ is the number of samples LoF variant B occurs in and $n$ is the total number of genomes.

A p-value for the co-occurrence of each gene pair was computed using this distribution. In order to account for multiple testing, actual false discovery rates were estimated by bootstrapping. The data in each row of the occurrence matrix was reshuffled: the frequency of each LoF was kept constant, but the genomes in which they occur were randomized. In this permuted data, no attractive or repellent relations exist between the LoFs, therefore, all LoF pairs picked up as either significantly over or under co-occurring are false positives. False discovery rates $r$ at different p-values cut-offs were estimated based on the number of false positives at each cut-off (Figure 3.3), averaged over repeated randomizations (5000 repeats):

$$r_{p<c} = \tilde{N}_{p<c}/N_{p<c}$$

where, at significance threshold $c$, $\tilde{N}$ is the average number of significant pairs identified in the randomized data and $N$ is the number of significant pairs identified in the original occurrence data.

An actual false discovery rate of 0.05 for both over and under co-occurrence was deemed acceptable. This corresponded to a cut-off of p = 0.0005 for over co-occurrence and p = 0.005 for under co-occurrence. This model identified 154 under co-occurring and 143 over co-occurring gene pairs, representing interactions between 90 LoF variants (Table 3.1).

### 3.3.2 Confounding Factors and Model Refinement

The aim of the analysis is to identify under co-occurring pairs. The assumption is that these pairs under co-occur in the healthy population because their combined effect is associated with a decreased probability of being healthy and therefore a decreased probability of the genome appearing in the thousand genome data. (It is worth noting that over co-occurrence could potentially correspond to alleviating genetic interactions where the presence of one LoF variant alleviates the negative effects of another. For simplicity, we will focus only on the under-occurrence in this work). However, genetic interactions are not the only reason some LoF variants my occur less often than expected: under co-occurrence may

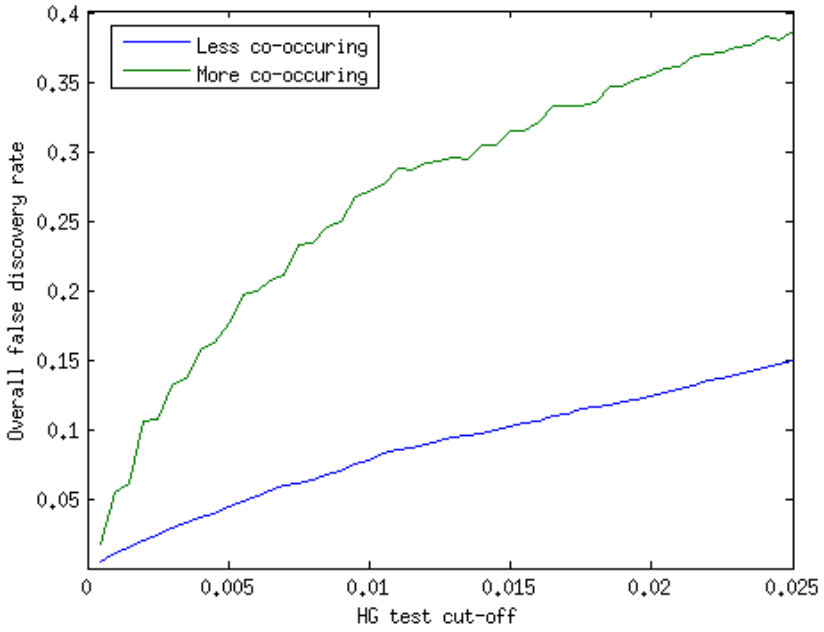


Figure 3.3: Correcting for multiple testing: the estimation of false discovery rates (y-axis) for p-value cut-offs from the hypergeometric (HG) test (x-axis). The figure shows total false discovery rate across all gene pairs for different p-value cut-offs when testing individual gene pairs, as estimated by bootstrapping.

also be due to linkage effects and population stratification.

### Linkage Effects

Genes that are located close together on a chromosome are more likely to be inherited together. Therefore, for genes on the same chromosome, the assumption of independent occurrence does not apply. If a population is descended from a relatively small ancestor population ('population bottleneck'), co-occurrence relations in that population may simply reflect those present by chance in the ancestor population, and thus not have functional importance. With independent inheritance, this effect would disappear rapidly, but a tendency to be inherited together will slow down this process.

Although linkage effects only act in one direction (*increasing* the probability of being inherited together), counter-intuitively, they can still result in under co-occurrence between LoF pairs. If a LoF-variant of gene A is located close to a non-LoF variant of gene B in the ancestor population, this will result in the over co-occurrence of LoF-A and non-LoF-B. However, if LoF-B occurs frequently enough, the over co-occurrence of LoF-A and non-LoF-B will appear as the under co-occurrence of LoF-A and LoF-B.

To assess the impact of linkage on the LoF pairs identified, we inspected pairs located on the same chromosome. 31 out of the 143 over co-occurring pairs and 16 out of 154 under co-occurring pairs were found on the same chromosome.

This is higher than we would expect by chance: sampling pairs of LoF variants randomly gives an average of approximately 10 pairs on the same chromosome. This suggests that some of the LoFs pairs we identify as over or under co-occurring may not be functionally significant, but simply due to linkage.

Because of chromosomal cross-over (the exchange of genetic material between homologous chromosomes) physically distant loci on the same chromosome are inherited independently. As a first approximation, we used a genomic average of recombination rate to estimate the distance at which gene pairs could be considered to be inherited independently. Yu et al [252] have estimated that for a distance of 0.75 mega base pairs (Mbp), the expected frequency of chromosomal cross-over is 1%. Thus, loci separated by over 37.5 ($0.75 * 50$) Mbp have 50% expected cross-over frequency and can thus be considered to be inherited independently. Therefore, as a rough guide, LoF pairs on the same chromosome separated by more than 40 Mbp should not be attributable to genetic linkage effects. More accurate estimates could also be made by considering chromosome specific recombination rate estimates.

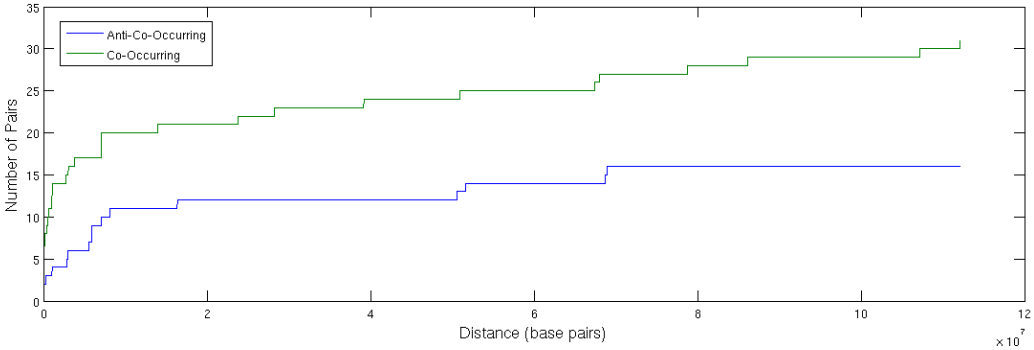


Figure 3.4: The number of LoF pairs in the significantly over and under co-occurring sets separated by less than $x$ base pairs. Pairs of LoF variants that are not on the same chromosome are ignored.

The distance at which loci can be considered independent could also be estimated from our data. Figure 3.4 shows the cumulative distribution of distance between the significantly over or under co-occurring LoF pairs. We can think of this distribution as arising from two processes: the underlying probability of observing over or under co-occurring LoFs, which, for now, we will assume is independent of distance (however, this point will be discussed further later) and the genetic linkage effect, which will decay with distance. The first process would be expected to give rise to an approximately linear relationship between the distance cut-off and the number of observed pairs (until the distance cut-off becomes of the same order of magnitude as chromosome length). Thus, above distances for which the genetic linkage effect becomes negligible, the cumulative distribution should be approximately linear. From our curve, this appears to be around 20 Mpb, an estimate of the same order of magnitude as the one based

on recombination rate.

It would also be possible to explicitly account for linkage effects in the null model used to identify significantly over or under co-occurring LoF pairs. However, there is evidence to suggest that genetic interactors are likely to occur close together on the genome. Firstly, functionally related gene pairs are known to be located closer together on the genome [176]. Secondly, gene duplication is known to be an important mechanism in the evolution of genetic interactions [104], increasing the probability of finding genetic interactors in close proximity to each other.

Thus, explicitly accounting for linkage effects in the null model or disregarding LoF pairs occurring too close together increases the risk of not detecting functionally interesting LoF pairs. Given the small size of the current sample, a more pragmatic strategy is to ignore the effects of linkage in the identification process, but consider them in the interpretation of the results.

### Population Stratification

The population sampled in the thousand genomes dataset is not genetically homogeneous. The Phase 1 data comprises genomes from people of African (n = 246), American (n = 181), East Asian (n = 286) and European (n = 379) decent. Physical separation, followed by genetic drift, can lead to systematic differences in allele frequency between populations, an effect referred to as population stratification. If two LoF variants occur at different frequencies in different populations, their under co-occurrence may simply be due to this population stratification and not reflect a functional interaction. Figure 3.5 illustrates the relative population specific frequencies of LoFs found to under co-occur with other LoFs. The uneven distribution of numerous LoFs in the different populations suggests that population stratification may indeed contribute to under co-occurrence.

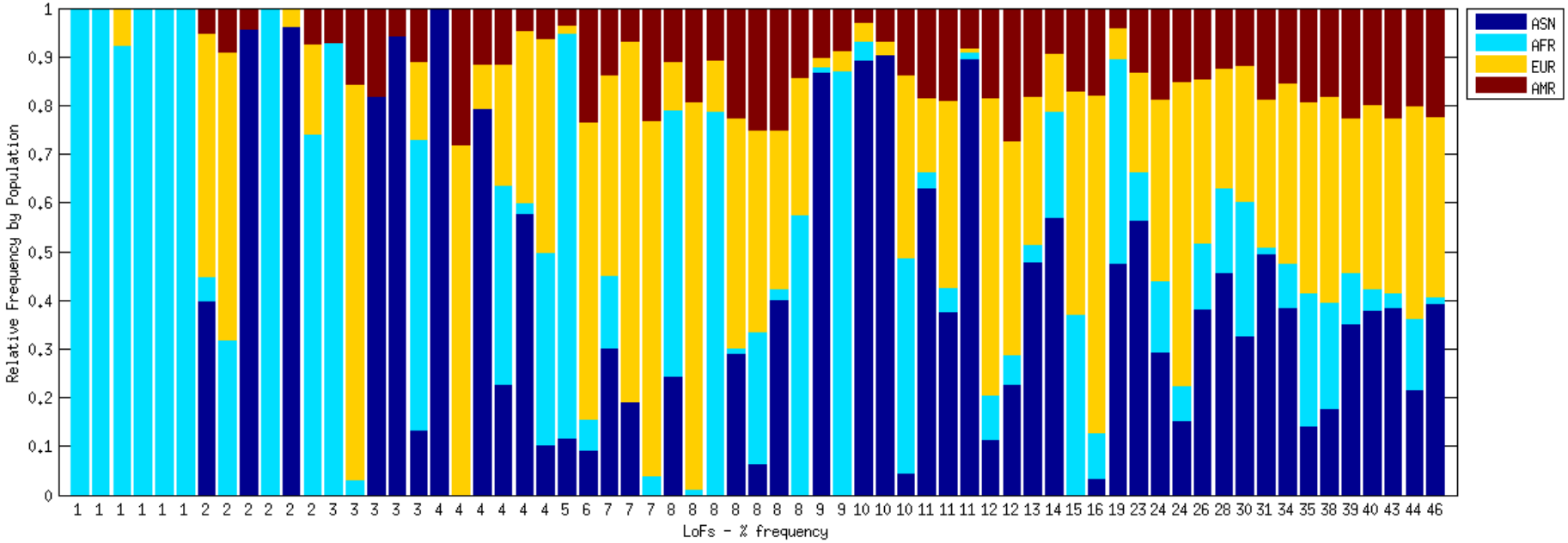


Figure 3.5: The frequency of LoF variants which show statistically signigicant under or over co-occurrence in the different populations. Each vertical bar represents a LoF variant, with the overall frequency of each LoF variant given on the x-axis. The bar is colored according the frequency of each LoF in the four populations. The purpose of the figure is to illustrate that the LoF variants may not be equally distributed among the populations. The names of the genes have been omited for clarity - the genes with significant under co-occurrence (i.e. putative genetic interactors) are listed in Table 3.1.

There are a number of ways in which we can attempt to discount this population effect. One solution would be to analyse each population separately - however, this will considerably reduce sample size and thus statistical power. It is therefore useful to consider alternative strategies. Theoretically it is possible to construct a hyper-geometric model which takes into account population structure, by considering co-occurrence within each population separately as illustrated in Figure 3.6. However, in practice, this distribution is prohibitively intensive to compute: if two LoFs occur in the same sample $k$ times, there are $4^k$ ways these occurrences could be distributed in the 4 populations ($k$ in African; $k-1$ in African, 1 in American; etc...).

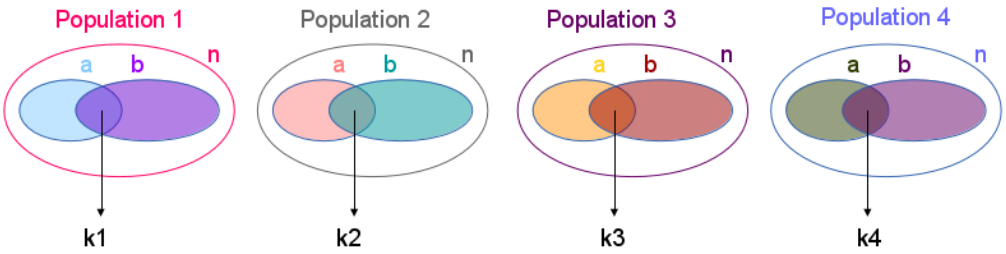


Figure 3.6: A model of co-occurrence taking into account population structure: each population is treated separately and total co-occurrence is the sum of the population co-occurrences. A Venn diagram depicts the samples ($n$) from each population, the samples ($a$) in which LoF variant A appears, the samples ($b$) in which LoF variant B appears and the samples in which the two variants co-occur.

A more practical approach is to estimate p-values by bootstrapping. As illustrated in Figure 3.7, by keeping the number of times each LoF occurs within a population constant, but randomizing the samples they occur in, a probability distribution can be computed and p-values estimated. An example of a distribution is shown in Figure 3.8.

False discovery rates were estimated as discussed in Section 3.3.1. The relationship between p-value cut-off and false discovery rate is illustrated in Figure 3.9. A total false discovery rate of 0.05 is achieved by selecting a significance threshold of $2.5 \times 10^{-4}$ for both under and over co-occurrence.

The disadvantage of this approach is that it may mask genuine genetic interactions: especially for low frequency LoF variants, the variants may be unevenly distributed between populations by chance. Such pairs would not be detected as significantly under co-occurring, because the effect would be attributed to population stratification.

It is also worth noting that each of the ancestry categories (i.e. African, American, East Asian and European) is divisible into sub-populations, although variation between these sub-populations is likely to be less significant than between the populations. Given that statistical power of our analysis is limited by the relatively small sample size, we chose to ignore sub-population structure.

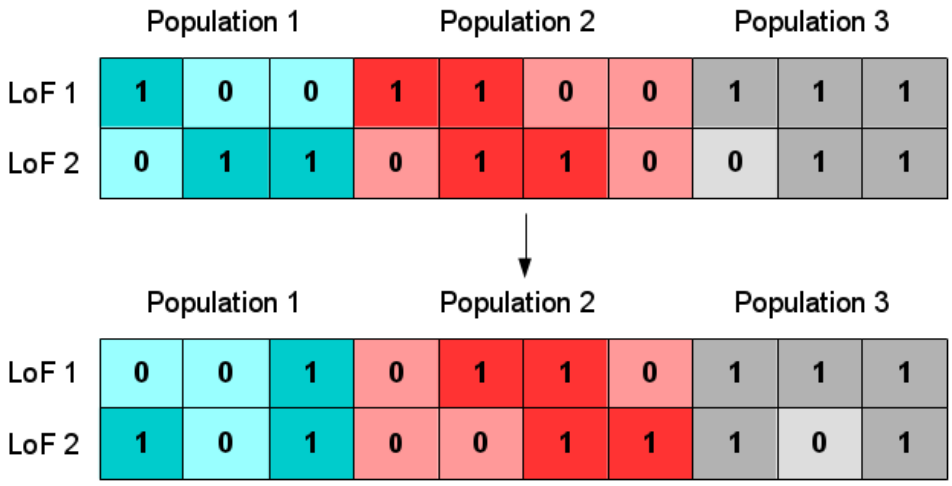


Figure 3.7: Bootstrapping for estimating p-values for the hypergeometric model taking into account population stratification. The frequency of each LoF variant is kept constant within each population, but the samples it occurs in are re-shuffled. Repeating this process allows estimating the distribution of expected co-occurrence, if all LoF variants behave independently.

### 3.3.3 Pairwise Interactions: Results

The original model identified 154 under co-occurring gene pairs, representing interactions between 65 LoF variants. 23 of these LoF variants were olfactory receptors (ORs). Loss of function in ORs is common in the human genome [143]. Based on prevalence of OR pseudogenes in different mammals, OR loss has become more common during primate evolution [55]. This has been interpreted as a decrease in the functional importance of OR genes. Thus OR genes may be more likely than other genes to be genuinely LoF tolerant. Although this does not preclude the existence of genetic interactions involving ORs, results involving ORs have been omitted for clarity.

Omitting pairs involving ORs left 68 pairs, corresponding to interactions between 37 LoF variants. These are listed in Table 3.1. In the population corrected model, only one non-OR pair was identified (a pair also identified in the original model).

### 3.3.4 Interpretation and Evaluation of Putative Interactions

Evaluating the reliability of our results is not straightforward. None of the putative genetic interactions from Table 3.1 overlap with documented genetic interactions in BioGRID. However, the genetic interaction data for human is very sparse: BioGRID only holds 1676 interactions for human (compared to 150394 for yeast). Indeed, out of the 317 LoF tolerant genes, only 8 had documented genetic interactions in BioGRID and none of these interactions involved another of LoF tolerant genes. Comparison with existing BioGRID genetic interactions

Table 3.1: Putative genetic interactions identified from the human genome data by testing for significant under co-occurrence. If the gene pair is on the same chromosome, the table also shows distance between the loci (in base pairs). One pair was also identified in a model correcting for population structure. This pair is italicised in the table.

| Gene 1 | Gene2 | Distance (bp) | Gene 1 | Gene2 | Distance (bp) |
|---|---|---|---|---|---|
| AC133919.6 | ALMS1 | - | C12orf60 | RP11-48B14.2 | - |
| ALMS1 | C17orf77 | - | C17orf77 | RP11-48B14.2 | 68994530 |
| AC133919.6 | C5orf27 | - | CYP2F1 | RP11-48B14.2 | - |
| C17orf77 | C5orf27 | - | DEFB126 | RP11-48B14.2 | - |
| AC018755.11 | FAM187B | 16377033 | DSCR8 | RP11-48B14.2 | - |
| ALMS1 | FMO2 | - | FMO2 | RP11-48B14.2 | - |
| AC018755.11 | FUT2 | 2889379 | FUT2 | RP11-48B14.2 | - |
| C5orf27 | FUT2 | - | GPR142 | RP11-48B14.2 | 68769581 |
| AC133919.6 | GAB4 | - | GRIA1 | RP11-48B14.2 | - |
| AC133919.6 | GDPD4 | - | HSD17B13 | RP11-48B14.2 | - |
| ALMS1 | GRIA1 | - | RP11-455G16.1 | RP11-48B14.2 | - |
| AC018755.11 | KRTAP13-2 | - | KRTAP4-8 | SLC22A14 | - |
| AC018755.11 | KRTAP4-8 | - | AC133919.6 | SLC35G6 | - |
| C5orf27 | KRTAP4-8 | - | FUT2 | SLC35G6 | - |
| IFNA10 | KRTAP4-8 | - | GRIA1 | SLC35G6 | - |
| *KRT37* | *KRTAP4-8* | *324724* | AC133919.6 | TEX26 | - |
| FUT2 | LILRA2 | 5891993 | C17orf77 | TEX26 | - |
| FUT2 | LILRA3 | 5597305 | AC133919.6 | ZNF284 | - |
| FUT2 | LILRB1 | 5936481 | C12orf60 | ZNF284 | - |
| AC133919.6 | MAGEE2 | - | CYP2F1 | ZNF284 | 2967891 |
| C17orf77 | MAGEE2 | - | FMO2 | ZNF284 | - |
| FUT2 | MAGEE2 | - | ALMS1 | ZNF474 | - |
| KRTAP4-8 | MAGEE2 | - | RP11-48B14.2 | ZNF474 | - |
| AC133919.6 | MAN2A1 | - | ZNF284 | ZNF474 | - |
| C17orf77 | MAN2A1 | - | AC133919.6 | ZNF804A | - |
| FUT2 | MAN2A1 | - | C17orf77 | ZNF804A | - |
| KRTAP13-2 | MAN2A1 | - | DEFB126 | ZNF804A | - |
| KRTAP4-8 | MAN2A1 | - | FMO2 | ZNF804A | - |
| C5orf27 | PKD1L2 | - | FUT2 | ZNF804A | - |
| MAGEE2 | PKD1L2 | - | GRIA1 | ZNF804A | - |
| MAN2A1 | PKD1L2 | - | ZNF474 | ZNF804A | - |
| KRTAP4-8 | PTCHD3 | - | ALMS1 | ZNF860 | - |
| ALMS1 | RP11-455G16.1 | - | RP11-48B14.2 | ZNF860 | - |
| AC133919.6 | RP11-48B14.2 | - | ZNF804A | ZNF860 | - |

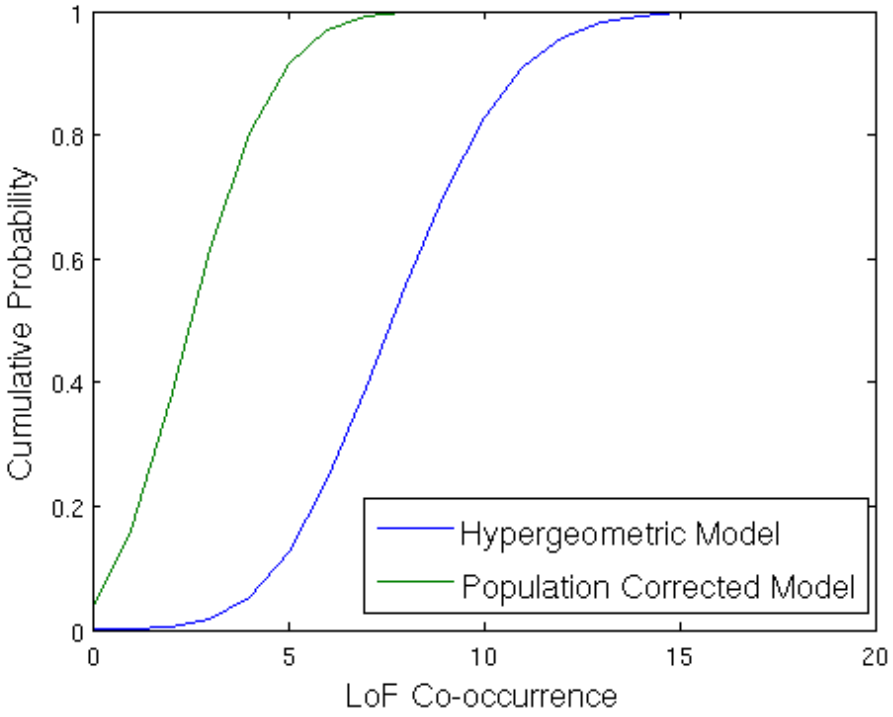


Figure 3.8: Example of a cumulative probability distribution for a LoF pair in the original hypergeometric model and in the population corrected model. One of the LoFs occurs in 45 samples (occuring in 26, 1, 16 and 2 Asian, African, European and American samples resectively) and the other in 58 samples (27, 13, 14 and 4 for Asian, African, European and American samples). For this LoF pair, the population corrected null model reduces the probability of co-occurrence of the LoF pair.

is therefore not an adequate way of validating our method. As an alternative approach to evaluating our results, we attempted to look for interactions between yeast orthologs of our putative interactors. However, only three of the putative interactors had yeast orthologs. Thus, we were unable to evaluate our results using existing interaction data.

Some of the putative interactions seem like plausible candidates. For example, in the pair identified using both the original and population corrected model (KRT37 and KRTAP4-8a), both proteins are involved in hair and nail formation (KRT37 is a keratin protein and KRTAP4-8a is a keratin associated protein). A functional association between these two proteins is therefore likely, although we found no evidence in the literature to indicate a potential genetic interaction between the two. It should also be noted that these two proteins are close together on the same chromosome: the result may therefore reflect a linkage effect.

The interaction between FUT2 (a Golgi stack membrane protein involved in antigen synthesis pathways) and the leukocyte immunoglobulin-like receptor group (LILRA2, LILRA3 and LILRB1) is also interesting. An association study in the Finnish population found a significant link between Celiac disease and

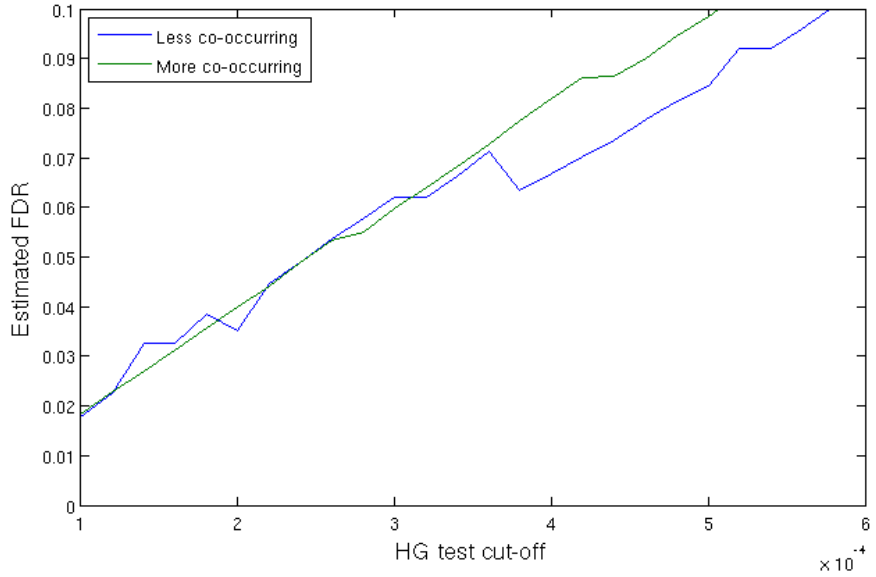


Figure 3.9: Estimation of false discovery rates for hypergeometric test significance threshold in the population corrected model. The figure shows total false discovery rate across all gene pairs for different p-value cut-offs when testing individual gene pairs.

FUT2 loss of function [173]. Several genome wide association studies on the other hand, have indicated a potential association between Celiac disease and the region of chromosome 19 containing the LIL receptors (region 19q13.4) [154]. It is worth noting that while FUT2 is also located on chromosome 19 (19q13.3), it is not located in the region identified in the genome wide association studies. It is therefore possible that we see the lower than expected co-occurrence of these loss of function variants because their co-occurrence increases the probability of developing Celiac disease. A caviat to this hypothesis is that because Celiac disease can develop at any age [51], the extent to which individuals with the disease would be excluded from participating in the 1000 genomes project is unclear.

## 3.4 Network Approaches to LoF pairs

The analysis above focuses solely on pair-wise relations between genes. However, genetic interactions may also arise between larger groups of genes. We are therefore interested in detecting under co-occurrence in larger communities of genes.

Modularity based clustering methods are a powerful way of detecting community structure in networks. This section will briefly introduce the use of modularity in community detection and then build on existing modularity based algorithms to make them applicable to epistatic community detection in genome data.

### 3.4.1 Introduction to Modularity

The identification of community structure in networks, or graph partitioning, has received a lot of attention in network science [160]. Broadly, approaches to network clustering problems can be divided into two categories depending on whether the number of clusters is pre-determined. When the number of clusters is pre-determined, partitioning the network can be expressed as a constrained optimization problem: assigning nodes into a fixed number of clusters such as to minimize the number of edges between clusters - there are a number of well known approaches that can be applied to this sort of problem. Unfortunately however, for a large number of situations in modern network science, the number of clusters is not pre-determined: alternative approaches are therefore required.

For a number of network science clustering problems, it is assumed that the network in question divides naturally into communities - the goal is thus to discover these communities from the network data [162]. Thus, instead of simply minimizing the number of edges between the clusters, partition methods seek to divide the network so that the number of edges running between clusters is *smaller than expected* [162]. To illustrate this difference, a partition seeking to minimize the number of edges between clusters, with no constraints on the number of clusters, would simply place all nodes into a single cluster. Taking into consideration the expected number of edges allows avoiding these types of solutions.

Thus, we seek a partition that maximizes a modularity function $Q$.

$$Q = (\text{number of edges within clusters}) - (\text{expected number of such edges})$$

The expected number of edges is computed using a null model representing a network without community structure. The choice of null model is thus extremely important in the computation of the partition. Newman proposed [162] a null model that preserves expected degree: the probability $P_{ij}$ of an edge being assigned between nodes $i$ and $j$ is proportional to the degree ($k$) of $i$ and $j$:

$$P_{ij} = k_i k_j / 2m$$

where $m$ is the number of edges in the network.

$Q$ is then given by:

$$Q = \frac{1}{2m} \sum_{ij} [A_{ij} - P_{ij}] \delta(g_i, g_j)$$

where $\delta(r, s) = 1$ if $r = s$ and 0 otherwise $g_i$ is the cluster into which node $i$ has been allocated. The factor $\frac{1}{2m}$ is irrelevant as it has no bearing on the solution of the optimization problem - it is included by convention, for historical reasons [162].

Considering first the division of the network into just two clusters, let $s_i = 1$ if node $i$ belongs to one cluster and $s_i = -1$ if node $i$ belongs to the other. Making use of $\sum_{ij} A_{ij} = \sum_{ij} Pij$, $Q$ can be written in terms of $s$ as:

$$Q = \frac{1}{4m} \sum_{ij} [A_{ij} - P_{ij}](s_i s_j + 1)$$

$$= \frac{1}{4m} \sum_{ij} [A_{ij} - P_{ij}](s_i s_j)$$

$$= \frac{1}{4m} \mathbf{s^T B s}$$

where $\mathbf{s}$ is a vector with $i^{th}$ element equal to $s_i$ and $\mathbf{B} = \mathbf{A} - \mathbf{P}$. Note that this is expression is similar to that used in spectral clustering, with $\mathbf{B}$ replacing the graph Laplacian. The reasoning below used to derive the partition is thus the same as that used in spectral clustering.

By expressing $\mathbf{s}$ as a linear combination of the normalized eigenvectors $u_i$ of $\mathbf{B}$ (so that $s = \sum_{i=1}^{n} a_i \mathbf{u_i}$ with $a_i = \mathbf{u_i^T} \cdot \mathbf{s}$), $Q$ can be re-written as:

$$Q = \frac{1}{4m} \sum_{i=1}^{n} a_i \mathbf{u_i^T B} \sum_{j=1}^{n} a_j \mathbf{u_j}$$

$$= \frac{1}{4m} \sum_{i=1}^{n} (\mathbf{u_i^T} \cdot \mathbf{s})^2 \beta_i$$

where $\beta_i$ is the eigenvalue corresponding to the eigenvector $\mathbf{u_i}$.

Thus, choosing $\mathbf{s}$ as the eigenvector corresponding to the largest eigenvalue would maximize $Q$. However, the elements of $s$ are constrained to take values of either 1 or $-1$. The best approximation is to set the value of $s_i$ according to the sign of the $i^{th}$ value of the eigenvector.

The division of the network into more than two parts is achieved through repeated divisions in two. However, treating the two clusters as separate networks would be incorrect: this would mean ignoring edges falling between the clusters, thus changing the degree of the nodes and therefore the modularity, leading to the maximization of the wrong quantity. Instead, the additional change in modularity $\Delta Q$ from subdividing cluster $g$ must be expressed explicitly, as outlined in [162]:

$$\Delta Q = \frac{1}{4m} [\sum_{i,j \in g} B_{ij}(s_i s_j) - \sum_{i,j \in g} B_{ij}]$$

Because $s_i^2 = 1$, $\sum_{i,j \in g} B_{ij}$ can be expressed as:

$$\sum_{i,j \in g} B_{ij} = \sum_{i,j \in g} (s_i s_j) \delta_{ij} \sum_{k \in g} B_{ik}$$

thus allowing $(s_i s_j)$ to be factored out and $\Delta Q$ to be written as:

$$\Delta Q = \frac{1}{4m} \sum_{i,j \in g} [B_{ij} - \delta_{ij} \sum_{k \in g} B_{ik}](s_i s_j)$$

$$= \frac{1}{4m} \mathbf{s^T B^{(G)} s}$$

where $B_{ij}^{(G)}$ is the $n_g \times n_g$ matrix corresponding to nodes belonging to cluster $g$, given by:

$$B_{ij}^{(g)} = B_{ij} - \delta_{ij} \sum_{k \in g} B_{ik}$$

If the additional contribution to the modularity from further sub-division of the cluster is positive ($\Delta Q > 0$, the partition is accepted. The sub-division process is continued until the modularity can no longer be increased by further division.

### 3.4.2 Anti-Community Clustering

As we have seen, the most positive eigenvalue and corresponding eigenvector contain information about the community structure of a network. Meanwhile, the most *negative* eigenvalue and corresponding eigenvector contain information about 'anti-community' structure: by using the eigenvector to determine clusters, we are minimizing the modularity, instead of maximizing it, thus selecting a partition where the number of edges within the cluster is smaller than expected [161]. Note that this is equivalent to reversing the sign of $\mathbf{B}$ and using the eigenvector corresponding to the largest eigenvalue.

### 3.4.3 Identification of Epistatic Communities from Co-Occurrence Data

In this section, we apply modularity based methods to identify higher order genetic interaction ('epistatic communities') between the LoF variants. Our goal is to divide the LoF variants into groups with lower than expected co-occurrence. First, we develop and test several variations on the modularity based methods outlined above, before applying these methods to identify groups of genes with genetic interactions.

#### Clustering Approaches

The first possible approach would be to treat the co-occurrence matrix (Figure 3.2) itself as the adjacency matrix of the network. Finding groups of genes with genetic interactions would thus correspond to partitioning the network into modules containing as few edges as possible.

This network could thus be clustered using the same modularity based method as above, using the degree preserving null model ($P_{ij} = k_i k_j / 2m$), where $k_i$ is

the weighted degree of node $i$ and $m$ is the total number of edges.

Another possibility is to compute the null model based on the frequency of the LoF variant in the thousand genomes data, i.e. explicitly modelling the expected co-occurrence between LoF variants if the probability of each variant being present in a sample was independent of the presence or absence of other variants:

$$P(i,j) = \frac{n_i n_j}{S}$$

where $S$ is the number of samples and $n_i$ is the number of samples LoF variant $i$ occurs in.

There is an important difference between these two null models: in the degree preserving null model, $\sum_{ij} A_{ij} = \sum_{ij} P_{ij}$, insuring the elements of $\mathbf{B}$ sum to zero. This, however, is no longer true when calculating the expectation based on the frequency of the LoF variants. Newman's model uses $\sum_{ij} A_{ij} = \sum_{ij} P_{ij}$ in the derivation of the expression for $Q$. Thus, for the frequency based null model, $Q$ should be expressed as:

$$Q = \frac{1}{4m} \sum_{ij} [A_{ij} - P_{ij}](s_i s_j + 1)$$

$$= \frac{1}{4m} [\sum_{ij} [A_{ij} - P_{ij}](s_i s_j) + \sum_{ij} [A_{ij} - P_{ij}]]$$

$$= \frac{1}{4m} [\mathbf{s^T B s} + \sum_{ij} B_{ij}]$$

Because $\sum_{ij} B_{ij}$ is a constant independent of the choice of partition, the result of the clustering will not be affected. With further sub-division of the modules, the constant will cancel out in the computation of $\Delta Q$. Thus, the modularity maximization/minimization algorithm is still valid.

However, although we can use the same algorithm for the clustering, the change in null model may have significant impacts on the number of modules we find: if, for example, the expected co-occurrence tends to be greater than the observed co-occurrence then the majority of the elements in $\mathbf{B}$ will be negative. Because the sign of the elements in $\mathbf{B}$ is what determines module allocation, all negative elements would lead to trivial solutions of all nodes in the same module or all nodes in different modules (depending on whether we are clustering according to smallest or largest eigenvector).

It is therefore worth exploring whether better results are achieved by introducing a weighting constant in the computation of $\mathbf{B}$ so that:

$$\mathbf{B} = \mathbf{A} - \alpha \mathbf{P}$$

where $\alpha = \sum_{ij} A_{ij} / \sum_{ij} P_{ij}$.

A third approach is to cluster the occurrence matrix $X$ (as defined in sec-

tion 3.2) instead of the co-occurrence matrix. The matrix can be thought to represent a bipartite graph, with LoF variants corresponding to one type of node and samples to the other. We aim to partition this network so as to minimize the number of edges falling between clusters. By extension of previous ideas, we can define a null model for this graph as:

$$P_{ij} = k_i k_j / 2m$$

where, as before, $k_i$ is the degree of node $i$ (i.e. the number of samples variant $i$ appears in) and $k_j$ is the degree of node $j$ (i.e. the number of LoF variants appearing in sample $j$). As before, we can use this to compute the modularity matrix $B$. Here, however, $X$ and therefore $P$ and $B$ are not square matrices. Indeed, we are clustering two types of nodes: genes and samples. Our expression for $Q$ therefore becomes:

$$Q = \frac{1}{4m}\mathbf{q^T B r}$$

where $\mathbf{q}$ and $\mathbf{r}$ are vectors indicating the module assignments of genes and samples respectively. As with $\mathbf{s}$, $q_i = 1$ if gene $i$ belongs to group 1 and $-1$ if it belongs to group 2.

Following the same reasoning as previously, we express $\mathbf{q}$ and $\mathbf{r}$ in terms of the left and right singular vectors of $\mathbf{B}$ to give:

$$Q = \frac{1}{4m}\sum_{i=1}^{n}\sum_{j=1}^{m}(\mathbf{u_i^T}\cdot\mathbf{q}) * (\mathbf{v_i^T}\cdot\mathbf{r})\sigma_{ij}$$

where $\mathbf{u_i}$ and $\mathbf{v_i}$ are the $i^{th}$ right and left singular vectors of $\mathbf{B}$ and $\mathbf{B} = \mathbf{U}\sigma\mathbf{V}$. Thus, as before, the optimal partition of the matrix is given by choosing $\mathbf{q}$ and $\mathbf{r}$ according to the sign of the elements of the right and left singular vectors corresponding to the smallest singular value.

This approach, however, becomes problematic when wanting to continue the subdivision of the network. Previously, modularity was re-expressed in terms of $B_{ij}^{(G)}$ relating to the sub-division of module $g$. The formulation of $g$ was dependent on $s_i^2 = 1$. This, however, is not necessarily true of $q_i * r_i$ and thus we cannot adapt our previous approach to the clustering of the original data matrix. In order to proceed, we can treat the modules as separate sub-networks, although this will lead to the optimization of the wrong quantity (see Section 3.4.1).

In summary, we have proposed four possible approaches for identification of epistatic communities:

1. Treating the co-occurrence matrix as a network and computing a modularity matrix using a degree preserving null model.

2. Treating the co-occurrence matrix as a network and computing a modularity matrix based on the frequency of occurrence of each LoF variant.

3. Modifying approach 2 to include a weighting in computing the modularity matrix to ensure its elements sum to zero.

4. Treating the original data matrix as a bipartite network and using a singular value decomposition to find the optimal partition.

### 3.4.4 Evaluation of Partition Approaches

Clustering approaches are typically benchmarked using data with a known community structure: the performance of the algorithm is evaluated based on how well the clustering replicates the real structure. However, real-word data with a known anti-community structure is not readily available. Therefore, to evaluate the performance of the four proposed algorithms, data with a predefined anti-community structure was created and the algorithms benchmarked using this data.

#### Creation of Simulated Data

Ideally, the data used to evaluate the clustering algorithms would have properties as close as possible to the real data. However, generating data with the correct properties is not trivial: there are a number of uncertainties in the processes governing the appearance of LoFs in genomes.

The approach we chose was to model the occurrence of LoF variants as independent variables (ignoring population and genomic distance effects) and introducing a term to model the decreased probability of finding certain groups of LoFs in the same healthy genome.

First, we create a set of $m$ pre-defined epistatic communities $C = (c_1, c_2, ..., c_m)$ by randomly assigning some of the LoF variants into one of the communities. (The specifics of the size and number of the communities used during testing are discussed below). Based on these communities, we define a matrix of interactions $J$, where $J_{ij} = 1$ if $i$ and $j$ are in the same community and 0 otherwise. We also assign each LoF variant $l_i$ a 'base' probability ($\lambda(l_i)$) of being present in the sample (where $l_i = 1$ indicates the presence of LoF $i$ and $l_i = 0$ indicates its absence). To create data similar to the true data, we set $\lambda(l_i)$ according to the frequency of occurrence of $l_i$ in the original data ($\lambda(l_i) = \mu_i$ if $l_i = 1$ and $\lambda(l_i) = 1 - \mu_i$, where $\mu_i$ is the frequency of LoF $i$).

We then model the joint probability distribution for a set of LoF variants $l = (l_1, l_2, ...l_n)$ being present in a genome as:

$$P(l) = \lambda(l_1) * \lambda(l_2) * ... * \lambda(l_n) * exp(-\sum_{i,j}(J_{ij} l_i l_j))$$

We sample the distribution using the Metropolis-Hastings algorithm:

1. The algorithm is initialized with a randomly selected sample $l$.

2. A candidate $l'$ for the next sample is generated by changing the absence of presence of a randomly selected LoF: $l'_i = 1 - l_i$

3. The candidate is accepted with probability $\alpha = P(l')/P(l)$. If $\alpha > 1$ the candidate is accepted automatically.

4. If the candidate is accepted, set $l = l'$.

5. Return to step 2 until the desired number of iterations is run.

The first sample is generated randomly. Consequently, the first samples generated by the algorithm will not follow the desired distribution. Figure 3.10 shows the LoF frequency in the first 7000 iterations. The algorithm appears to reach equilibrium around $n = 1000$. We therefore discarded the first 1500 iterations.

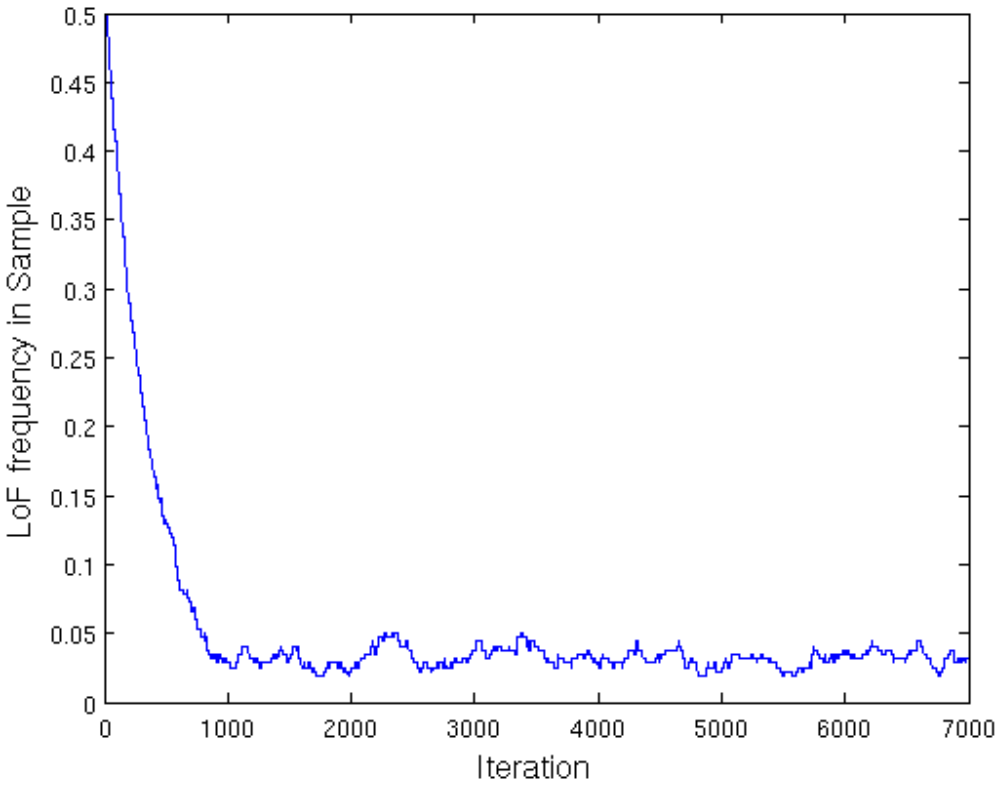


Figure 3.10: Figure representing the convergence of the Metropolis-Hastings algorithm. The plot show the frequency of LoF variants in samples generated from the first 7000 iterations of the algorithm. The frequency stabilizes around 1000 iterations, suggesting the algorithm has converged to the desired distribution.

The samples generated by successive iterations of the Metropolis-Hastings algorithm are highly correlated. To correct for this, samples were only taken every 500 iterations. A total of 1000 samples were generated.

This model is an ad-hoc method of creating the testing data and does not replicate the true data perfectly: $\lambda$ is the empirical distribution we seek to recreate, but the inclusion of the interaction term changes the expected frequency of the LoF variants. Furthermore, LoF variants which do not occur in any sample are not included in the original data. Thus, LoF variants with a frequency of 0 in the simulated data were removed.

Figure 3.11 compares distributions for the number of samples a LoF appears in (variant LoF frequency) and the number of LoF variants occurring in a sample (sample LoF frequency). The distributions for the original and simulated data are similar in shape, although the scaling is different, due to the removal of LoF variants with a frequency of 0 from the simulated data.

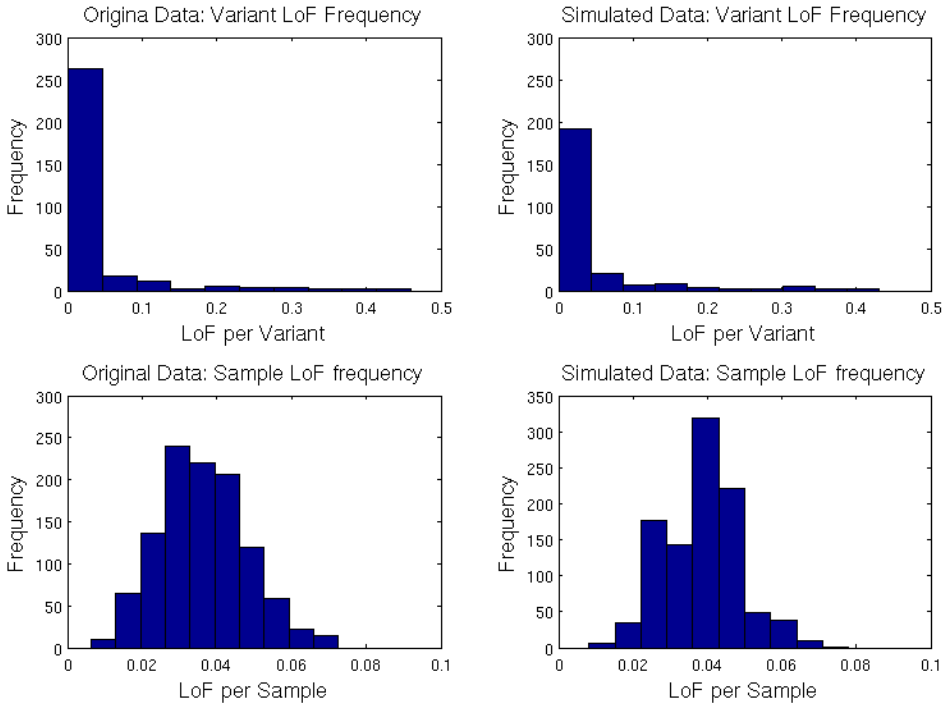


Figure 3.11: Comparison of distributions for the number of samples a LoF appears in (variant LoF frequency) and the number of LoF variants occurring in a sample (sample LoF frequency). The values on the y-axis for the variant LoF frequencies are different, because variants with a frequency of zero have been removed.

### Evaluation Results

In order to evaluate the performance of the four methods, we require a metric capturing how well the set clusters generated by the algorithm ($\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$) correspond to the communities present in the simulated data ( $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$). Our clustering algorithms are not guaranteed to divide the nodes into the same number cluster - we therefore select to use normalized mutual information (NMI)

and Rand Index (RI) because of their suitability for assessing methods giving rise to differently sized clusters [146].

Normalized mutual information is a measure with an information theoretic interpretation: mutual information $I$ captures the extent to which information about cluster membership tells us about class membership. If the number of objects to cluster (in our case, the number of nodes in the network) is $N$, mutual information is given by:

$$I = \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|}$$

If the cluster assignment is random - i.e. knowing which cluster a node has been assigned to gives no information about which class it might belong to, $I$ will be 0. $I$ is maximized when the clustering assignment corresponds to the exact classes. However, because there is no penalty for further sub-division, $I$ is also trivially maximized by assigning each node into its own cluster. $I$ would thus exhibit a bias favouring partitions with numerous clusters. To correct for this, we normalize $I$:

$$I_{norm} = \frac{2I(\Omega;\mathbb{C})}{[H(\Omega) + H(\mathbb{C})]}$$

where $H(\Omega)$ and $H(\mathbb{C})$ are give by:

$$H(\Omega) = -\sum_k \frac{|\omega_k|}{N} \log \frac{|\omega_k|}{N}$$

$$H(\mathbb{C}) = -\sum_j \frac{|c_j|}{N} \log \frac{|c_j|}{N}$$

$I_{norm}$ takes on values between 0 and 1 and, because $H(\Omega)$ tends to also increase with the number of clusters, is less sensitive to the number of clusters.

| | Same Community | Different Community |
|---|---|---|
| Same Cluster | TP | FP |
| Different Cluster | FN | TN |

Table 3.2: Illustration of true and false positives and negatives in the assessment of clustering algorithms

The Rand Index measures the proportion of node pairs that have been correctly assigned. As illustrated in Table 3.2, there are two types of correctly assigned nodes:

1. true positives (TP: pairs belonging to the same community having been correctly assigned to the same cluster), and

2. true negatives (TN: pairs belonging to different communities having been correctly assigned into different clusters).

and two types of incorrectly assigned nodes:

1. false positives (FP: pairs belonging to different communities having been assigned to the same cluster), and

2. false negatives (FN: pairs belonging to the same community having been assigned into different clusters).

The Rand Index is then given by:

$$RI = \frac{TP + TN}{TP + TN + FP + FN}$$

We tested the four clustering algorithms on simulated data containing communities of different sizes: small (63 communities of 5 nodes), medium communities (31 communities of 10 nodes), large communities (10 communities of 30 nodes) and very large communities (3 communities of 100 nodes). Method 2 clearly outperforms the other clustering methods on both NMI (Figure 3.12) and RI (Figure 3.13).

To control for the potential effect of the number of clusters found by each algorithm on the evaluation metric, we compare the cluster assignment to a random cluster assignment with the same number of clusters and nodes per cluster. The results are presented in Table 3.3. Although all methods perform better than random, the performance is not particularly high in absolute terms, especially as measured by NMI. Furthermore, all methods generated, on average, below 5 clusters. This will potentially limit the usefulness of these clustering methods for identifying real epistatic communities.

Method 2 outperforming the other clustering approaches is not unexpected - it is most probably a consequence of a more astute choice of null model for the expected co-occurrence. Method 1 generates the expected co-occurrence using a degree preserving model. The degree preserving model assumes the degree of a node represents an inherent property of the node: its interaction probability. Edges between nodes are generated according to the joint probability distribution of the node interaction properties. This does not make sense for the co-occurrence network: inherent properties of the LoFs are captured by their frequency (i.e. total *occurrence*), not total co-occurrence.

It is also worth noting that method 2 models the expected co-occurrence as the joint probability of independent variables based on their observed frequency. This is also how to joint probability of the LoFs is modelled during generation of the simulated data, prior to multiplication by the interaction term. Thus, it could be argued that method 2 does well because they way it models the data is also the model used in generating the test data. It is difficult to avoid this problem: the data generation models a very plausible mechanism for the processes generating the original data. However, it would be interesting to check the performance of the algorithms on data generated using a different mechanism.

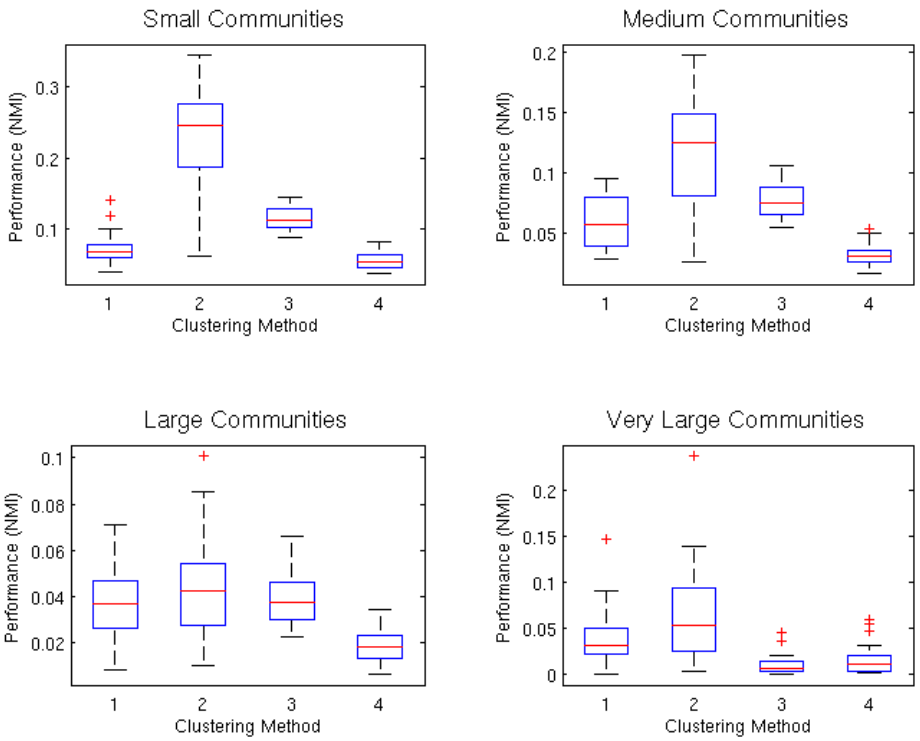


Figure 3.12: Comparative performance of the four clustering algorithms on simulated data containing different size communities as measured by NMI. **Method 1**: clustering the co-occurrence matrix using Newman modularity; **Method 2**: clustering the co-occurrence matrix, computing modularity using frequency based expectation; **Methods 3:** similar to method 2, but ensuring the modularity matrix sums to 0; **Method 4**: using modularity-type clustering to cluster the original data. Refer to section 3.4.3 for further details. The communities present in the data were as follows: small (63 communities of 5 nodes), medium communities (31 communities of 10 nodes), large communities (10 communities of 30 nodes) and very large communities (3 communities of 100 nodes).

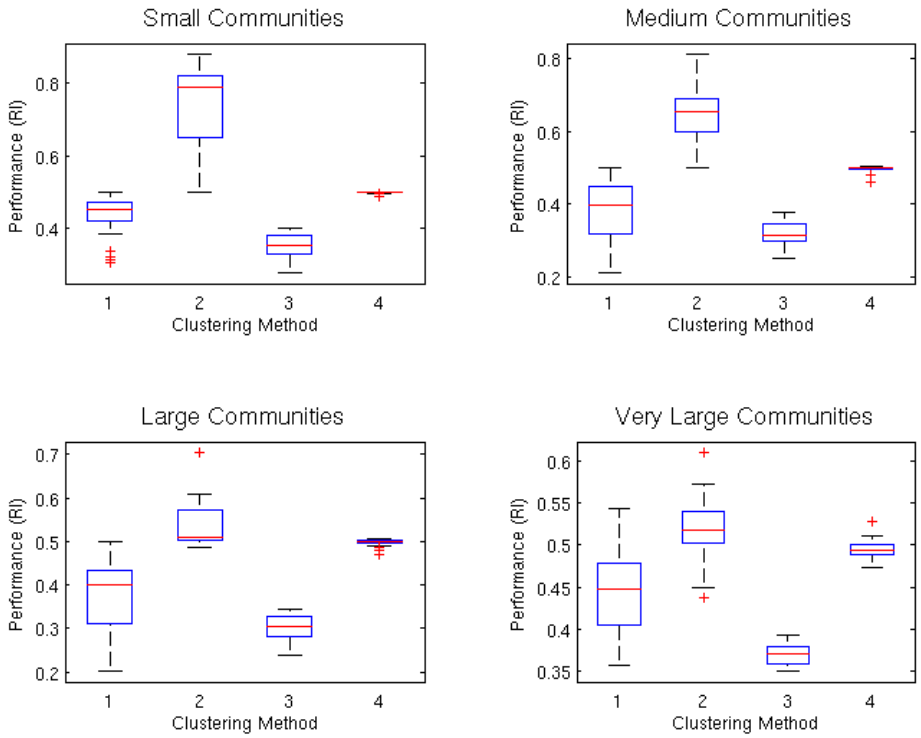


Figure 3.13: Comparative performance of the four clustering algorithms on simulated data containing different size communities as measured by RI. **Method 1**: clustering the co-occurrence matrix using Newman modularity; **Method 2**: clustering the co-occurrence matrix, computing modularity using frequency based expectation; **Methods 3:** similar to method 2, but ensuring the modularity matrix sums to 0; **Method 4**: using modularity-type clustering to cluster the original data. Refer to section 3.4.3 for further details. The communities present in the data were as follows: small (63 communities of 5 nodes), medium communities (31 communities of 10 nodes), large communities (10 communities of 30 nodes) and very large communities (3 communities of 100 nodes).

| Community Size (nodes) | Number of Communities | Clustering Method | Number of Clusters | NMI | NMI (random) | P-value | RI | RI (random) | P-value |
|---|---|---|---|---|---|---|---|---|---|
| 5 | 63 | 1 | 2.320 | 0.073 | 0.014 | 0.000 | 0.435 | 0.363 | 0.000 |
| 5 | 63 | 2 | 6.520 | 0.230 | 0.024 | 0.000 | 0.749 | 0.571 | 0.000 |
| 5 | 63 | 3 | 4.600 | 0.115 | 0.009 | 0.000 | 0.353 | 0.370 | 0.002 |
| 5 | 63 | 4 | 2.000 | 0.055 | 0.013 | 0.000 | 0.500 | 0.419 | 0.002 |
| 10 | 31 | 1 | 2.960 | 0.058 | 0.007 | 0.000 | 0.380 | 0.341 | 0.000 |
| 10 | 31 | 2 | 4.360 | 0.116 | 0.012 | 0.000 | 0.648 | 0.552 | 0.000 |
| 10 | 31 | 3 | 4.160 | 0.077 | 0.010 | 0.000 | 0.322 | 0.253 | 0.000 |
| 10 | 31 | 4 | 2.000 | 0.032 | 0.003 | 0.000 | 0.497 | 0.453 | 0.000 |
| 30 | 10 | 1 | 2.760 | 0.037 | 0.002 | 0.000 | 0.376 | 0.356 | 0.000 |
| 30 | 10 | 2 | 2.600 | 0.046 | 0.001 | 0.000 | 0.536 | 0.511 | 0.000 |
| 30 | 10 | 3 | 3.720 | 0.040 | 0.003 | 0.000 | 0.303 | 0.309 | 0.000 |
| 30 | 10 | 4 | 2.000 | 0.019 | 0.001 | 0.000 | 0.497 | 0.482 | 0.000 |
| 100 | 3 | 1 | 2.640 | 0.042 | 0.001 | 0.000 | 0.447 | 0.432 | 0.000 |
| 100 | 3 | 2 | 2.080 | 0.064 | 0.000 | 0.000 | 0.519 | 0.485 | 0.000 |
| 100 | 3 | 3 | 2.200 | 0.010 | 0.001 | 0.000 | 0.370 | 0.377 | 0.000 |
| 100 | 3 | 4 | 2.000 | 0.016 | 0.000 | 0.000 | 0.495 | 0.485 | 0.000 |

Table 3.3: Table of results for the comparison of the clustering algorithms on simulated data. The values represent the average of 25 simulated dataset. Reported p-values are derived from a two-tailed t-test.

### 3.4.5 Epistatic Communities

Based on the results on simulated data, method 2 was used to cluster the co-occurrence matrix into putative epistatic communities, including (Table 3.4) and excluding (Table 3.5) the olfactory receptors. Interestingly, on this data, the number of clusters was considerably larger than on the simulated data - this suggests that the simulated data does not fully capture all the properties of the original data.

In order to examine whether the clusters correspond to particular biological functions, we performed a GO-enrichment analysis on each cluster (using the clusters excluding the OR receptors).

Enrichment analysis is a method of determining whether a specific feature - in this case, GO-category - is significantly over- or under-represented in a gene list, compared to a background gene list. All enrichment analyses presented here were performed using GO::Term-Finder [22], which computes p-values using the hypergeometric distribution:

$$p = 1 - \sum_{i=0}^{k-1} \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{i}}$$

where $N$ is the total number of genes in the background list, $M$ is the number of genes with a given annotation in the background list, $n$ is the size of the gene list of interest and $k$ is the number of annotated genes in the gene list of interest.

P-values were corrected using Bonferroni correction for multiple hypothesis testing.

The analysis gave significant enrichment in only one of the cluster: the cluster AC079612.1, C2orf91, CAPN9, MAGEE2, TCHHL1, TTC28, UNC93A was enriched for the label 'unannotated' (corrected p-value: 0.00417).

The lack of coherent GO annotations within the clusters does not necessarily indicate a poor quality clustering. Firstly, because all the genes are loss of function tolerant, it is plausible they may be less well studied than other genes with a clearer impact of phenotype. Thus, the lack of functional enrichment may not reflect a lack of functional coherence, but a lack of knowledge about the genes in question. This would also explain the enrichment for the 'unannotated' label in one of the clusters. Secondly, the number of genes studied is small - a greater number of genomes would contain a greater number of LoF variants, potentially making it easier to identify functionally coherent groups of LoF tolerant genes.

## 3.5 Conclusion

The central idea motivating the work in this chapter is the unexpected frequency of loss of function variants in healthy genomes. One of the possible explanations is that genes appearing to be tolerant of loss of function are conditionally essential: loss of function in pairs or groups of genes may be tolerated if they occur

Table 3.4: Epistatic communities identified using modularity based clustering of the co-occurrence matrix.

| |
|---|
| AC018755.11 AC092171.1 AKR1E2 C12orf60 C17orf107 C3orf49 C5orf49 CALHM2 CCDC7 CD207 COL23A1 CSTL1 DCLRE1A FTHL17 GAB4 GPRC6A IFNE LRRC39 MAGEB16 OR10C1 OR4L1 OR51H1P OR51V1 OR52A1 OR5M1 OR5M10 OR5M11 OR7G3 OR8I2 OR9K2 PNLIPRP3 POM121L4P PRB4 RAET1E RP11-48B14.2 RP11-794P6.2 SDR42E1 SPERT TNK1 TTC24 |
| CEACAM4 ZNF469 |
| ABCC12 ACSM3 C19orf71 C8orf44 CBLC DBF4B EXO5 FAM187B FBXL21 FMO2 FUT2 GDPD4 GPR142 GRIA1 H2BFM MBL2 MRGPRX3 MST1R NOX5 OR10D3 OR2T4 OR52M1 OR5AC2 OR5H15 OR8B3 PLA2G4D PSORS1C2 RESP18 RFPL1 RP11-113D6.6 RP11-65D24.2 RP11-830F9.6 SDIM1 TAS2R8 TBC1D29 TCHHL1 TIGD6 UGT2B10 ZAN ZNF681 ZNF80 |
| OR4C16 SEMA4C |
| ARID3A C2orf91 MAN2A1 OR10R2 PKD1L2 |
| C14orf180 C17orf77 C17orf97 DEFB126 EBF4 FLJ43860 KHDC1L LPA OR10A6 OR52I2 OR52K2 SLFN12L SNX31 SPATA4 TRIM22 UBE2NL UMODL1 |
| CELA1 FADS6 IFNA10 OR2G6 OR4C11 PLA2R1 TLR5 TRIM73 TXNRD3NB ZC2HC1C |
| AC079612.1 APOBEC3B C9orf43 CLYBL DDIT4L GPR135 HTN3 KRT37 LIPJ MS4A12 OLFM4 OR2A5 PRAMEF4 RP11-766F14.2 TTC28 |
| ABHD14B AC132186.1 AC133919.6 ACTR3C AGAP6 AHCTF1 ALMS1 C10orf113 C14orf182 C18orf56 C4orf17 COX6B2 CR392000.1 CST9 CTD-2373H9.6 DSCR8 EIF3CL GSTT2 GSTT2B ITIH5 JMJD1C KRTAP1-1 KRTAP13-2 KRTAP4-7 KRTAP4-8 KRTAP9-1 LAD1 NRAP OR10AD1 OR10G7 OR2C1 OR2D3 OR2T11 OR2T27 OR2V2 OR4S2 OR4X2 OR52B4 OR5AR1 OR5H1 OXGR1 PLEKHG5 RP11-276H1.3 RP11-455G16.1 RP11-481A20.11 SATL1 SCN8A SPATA31A6 SPZ1 TGFB1 TMEM82 UBQLNL UGT2B28 UTS2D ZFP91 ZFP91-CNTF ZNF790 |
| BPIFB3 COL6A5 DKFZP779J2370 LILRA2 OR52N4 OR6C74 PSG1 SLC35G6 |
| C2orf57 GLT6D1 IDO2 OR5B17 TMEM198 |
| FAM111B LILRA3 OR51I2 PXDNL |
| AC129492.6 AL359878.1 ATP13A5 C13orf45 C6orf123 CAPN9 CFHR1 CRIPAK CYP2A13 CYP2F1 HBM HID1 HSD17B13 IDI2 KRT31 KRTAP1-5 METTL7B MOGAT1 NACA2 NOXO1 OR13C2 OR3A1 OR4D6 OR4X1 OR51F1 OR51I1 OR6C4 OR6Q1 OR7G1 PSG9 RETNLB SLC22A14 SLCO1B1 SMUG1 SPTBN5 TEX22 TSPAN19 VN1R1 ZNF417 ZNF474 ZNF804A |
| OR4P4 RAI1 RP11-297N6.4 |
| FAM25A GEN1 PTCHD3 ZNF284 |
| ABCA8 C11orf21 C3orf14 CPN2 DEFB128 DKFZP434O1614 DNAH8 ENPP7 GBP7 IL34 LCE4A OR10X1 OR2D2 OR5K4 OTOP1 PCDHA3 PKHD1L1 PTGDR RHD SOX13 STK19 TAS2R46 TAS2R7 ULBP3 ZNF812 |
| CENPBD1 CYP2C18 OR13D1 PCDHGA8 SERPINB3 TLR10 |
| CYP2D6 |
| LILRB1 NIPA2 UNC93A ZNF860 |
| C10orf68 CYP2C19 GRIN3B MAGEE2 OR11G2 OR13C4 OR6C1 |
| C5orf27 |
| C1orf227 DNAJC28 NT5C1B-RDH14 OR1B1 RFX8 TOR1AIP1 ZNF154 |
| TAAR2 |
| ARMS2 OR2B11 RP1L1 TEX26 |
| C21orf88 COL16A1 OR51Q1 PP12708 TRIM38 |
| CD200R1 OR4D10 |

Table 3.5: Epistatic communities identified using modularity based clustering of the co-occurrence matrix (excluding olfactory receptors).

| |
|---|
| BPIFB3 DNAH8 FBXL21 GEN1 PSG9 RAET1E RP11-766F14.2 RP11-830F9.6 SDIM1 TAAR2 TRIM38 VN1R1 ZNF284 ZNF417 |
| ABCA8 ABCC12 AC092171.1 AC129492.6 ACSM3 AKR1E2 ARID3A C13orf45 C17orf97 C18orf56 C19orf71 CCDC7 CELA1 COL16A1 CYP2A13 DCLRE1A DDIT4L DEFB128 DNAJC28 GRIA1 HBM ITIH5 KRT37 LAD1 MBL2 MOGAT1 PLEKHG5 PRAMEF4 PRB4 PSORS1C2 PTCHD3 PTGDR RESP18 RP11-276H1.3 SATL1 SLC35G6 SLFN12L SNX31 SOX13 SPATA4 SPERT TAS2R7 TAS2R8 TBC1D29 TGFB1 TLR10 TSPAN19 UBE2NL UMODL1 ZAN |
| AC018755.11 AC132186.1 ACTR3C ATP13A5 C11orf21 C17orf77 C1orf227 C21orf88 C2orf57 C3orf14 C4orf17 C5orf27 C6orf123 CALHM2 CENPBD1 CFHR1 CLYBL COL23A1 COX6B2 CST9 CSTL1 CYP2C19 CYP2F1 DBF4B DEFB126 DKFZP434O1614 DKFZP779J2370 EBF4 EXO5 FAM25A FUT2 GBP7 GSTT2B HTN3 IDI2 IFNE JMJD1C KRT31 KRTAP1-1 KRTAP1-5 KRTAP9-1 LCE4A LILRA2 LILRA3 LRRC39 MAGEB16 METTL7B NACA2 NIPA2 NOXO1 NRAP NT5C1B-RDH14 OLFM4 PCDHA3 PCDHGA8 PLA2G4D PNLIPRP3 POM121L4P PXDNL RETNLB RHD RP11-297N6.4 RP11-455G16.1 RP11-481A20.11 RP11-794P6.2 SLCO1B1 SPTBN5 SPZ1 STK19 TEX22 TRIM73 TXNRD3NB UBQLNL UGT2B10 UGT2B28 UTS2D ZNF154 ZNF681 ZNF790 ZNF804A ZNF812 |
| C8orf44 DSCR8 FADS6 HID1 NOX5 OTOP1 PKD1L2 SMUG1 TNK1 TRIM22 ZNF80 |
| C3orf49 |
| SLC22A14 ZC2HC1C |
| ABHD14B AGAP6 AHCTF1 COL6A5 FAM111B KRTAP13-2 MAN2A1 OXGR1 PLA2R1 RAI1 SPATA31A6 TMEM82 TTC24 ZNF469 |
| ALMS1 CR392000.1 FLJ43860 FTHL17 GAB4 KHDC1L KRTAP4-8 LIPJ LPA MRGPRX3 RP11-65D24.2 TLR5 TOR1AIP1 |
| PP12708 |
| AC079612.1 C2orf91 CAPN9 MAGEE2 TCHHL1 TTC28 UNC93A |
| AL359878.1 APOBEC3B C10orf68 C14orf180 C14orf182 C17orf107 CD200R1 CD207 CRIPAK CYP2C18 EIF3CL FMO2 GPR135 GPRC6A GSTT2 IDO2 KRTAP4-7 LILRB1 MS4A12 MST1R PKHD1L1 |
| CTD-2373H9.6 GLT6D1 GRIN3B PSG1 |
| C9orf43 GDPD4 GPR142 RFX8 TAS2R46 |
| AC133919.6 ARMS2 C10orf113 C12orf60 C5orf49 CBLC CEACAM4 CPN2 CYP2D6 ENPP7 FAM187B H2BFM HSD17B13 IFNA10 IL34 RFPL1 RP11-113D6.6 RP11-48B14.2 SDR42E1 SERPINB3 TMEM198 |

separately, but the combined effect of these loss of function variants on the same genome is deleterious. This is known as genetic interaction.

We would expect LoF variants of genetic interactors to occur together on healthy genomes less often than expected by chance. In this chapter, we develop methods for detecting pairs or communities of genetic interactors and use them to identify putative genetic interactors in data from the 1000 genomes project. We attempt to validate the genetic interactions we identify using known genetic interactions in human. While none of the identified gene pairs correspond to known interactors, this does not necessarily indicate our approach is flawed: genetic interaction data for humans is sparse - it is therefore not unexpected that our gene pairs did not overlap with known interactors.

We also attempt to validate the putative interactions by considering what is known about the function of the gene pairs. Some of the pairs we identified seem promising: particularly, we identified interactions potentially relating to Celiac disease. There are two caveats to this observation. Firstly, as Celiac disease can manifest at any age, it is unclear whether sufferers would necessarily be excluded from the 1000 genomes dataset. Secondly, the gene pairs potentially representing interactions associated with Celiac disease were all relatively close together on the same chromosome. The interaction could therefore simply arise due to linkage effects instead of representing a genuine functional association between the genes.

Fundamentally, the work in this chapter is limited by the relatively small sample of genomes. Indeed, the main value of the work presented here is the method development, rather than the biological insight generated from this dataset. However, genome availability is increasing rapidly - the ideas developed in this chapter could therefore be applied on a larger dataset once one becomes available.

# Chapter 4

# Functional Association Networks For Prediction of Loss of Function Tolerance

This chapter explores the use of network data in predicting the functional consequences of loss of function mutation. We extend previous work on using network centrality as an indicator of functional importance by integrating this approach with the kernel-based guilt-by-association prediction methods explored in Chapter 2. We show that integration of the guilt-by-association approach improves the performance beyond using centrality only.

## 4.1 Introduction

Predicting the functional impact of genomic variation is a key challenge for computational biologists. The falling cost of personal genomes makes identifying potential disease variants of great clinical interest. Prediction is also interesting from a basic science point of view: the predictive power of our models is a measure of how well they represent cellular function and organisation.

The impact of variation in a protein coding gene can be thought of at different levels of resolution: the effect on a protein's primary sequence, on its three dimensional structure, on its interactions with other molecules, and finally, on the cell and organism as a whole. Ultimately, comprehensive understanding and prediction of the effects of variation will require integration of information across all these levels of resolution.

In this work, we are interested in the functional effects of complete homozygous loss of function in individual genes. Networks are a natural tool to assessing the wider consequences of the loss of an individual protein. Network approaches can be applied to the problem in two distinct ways:

1. Network centrality: essential genes tend to be more central in protein interaction networks [102] while loss of function tolerant genes tend to

have lower centrality [112]. Network centrality is therefore often used as a feature when attempting to categorise the functional impact of a variant.

2. Guilt-by-association: previous work suggests that the effect of loss of function in a gene can be predicted from the functional effects of its neighbours in a PPI network [90]. Guilt-by-association approaches could therefore be applicable to predicting the consequences of variation.

Previous work has mainly focused on PPI networks - these have been applied to predicting mutations that may act as cancer drivers [112], or discrimination between haploinsufficiency (where a single non-functional allele causes disease - i.e. dominant disease genes) and haplosufficiency (a single non-functional allele does not cause disease - i.e. recessive disease genes and loss of function tolerant genes) [90]. Integrated networks, comprising information from PPI, phosphorylation, metabolic, signalling, genetic and regulatory networks have also been utilised to discriminate between LoF tolerant and essential genes [111].

In this section, we seek to extend previous work by building a 3 class predictor discriminating between LoF tolerant, recessive and dominant disease genes, using PPI, genetic interaction and metabolic networks, as well as guilt-by-association methods. Furthermore, we explicitly examine the relationship between centrality and functional significance in different networks. Because different networks hold different representations of functional association, the correlation between essentiality and centrality may not hold in different networks.

## 4.2 Datasets

The dominant and recessive disease gene sets used in this work were obtained from the Online Medelian Inheritance in Man (OMIM) database [77] by text mining. The LoF tolerant gene set is the set identified from the 1000 genomes project (see Chapter 3).

The PPI network was downloaded from the interaction repository BioGRID [217], the genetic interaction network derived from a radiation hybrid screen [138] (see Section 4.3 for details), the metabolic network was downloaded from the Recon 2 database [226] and the functional association network from the STRING database [101]. Table 4.1 gives the number of genes from each gene class (dominant, recessive and tolerant) present in each of the networks.

## 4.3 Network properties

First, the network properties of three gene classes were explored in different types of network. If significant differences between the dominant, recessive and LoF tolerant genes can be found, these properties can be used in building a predictive tool for discriminating between the classes.

| | Dominant | Recessive | Tolerant |
|---|---|---|---|
| All genes | 298 | 456 | 317 |
| Physical | 285 | 420 | 112 |
| Genetic | 268 | 423 | 131 |
| Metabolic | 27 | 198 | 14 |
| Functional Association | 251 | 395 | 152 |

Table 4.1: The number of the genes from the 3 categories (dominant disease gene, recessive disease gene and loss of function tolerant gene) present in each of the networks used for prediction.

### 4.3.1 Protein Interaction Networks

In PPI networks (from BioGRID [217]), as expected degree and betweenness centrality varied according to functional impact: dominant disease genes were more central than recessive disease genes, which were, in turn, more central than LoF tolerant genes (Figure 4.1). Differences between all three categories are significantly different (Wilcoxon ranked sum, $p < 10^{-6}$ for degree, $p < 10^{-4}$ for centrality).

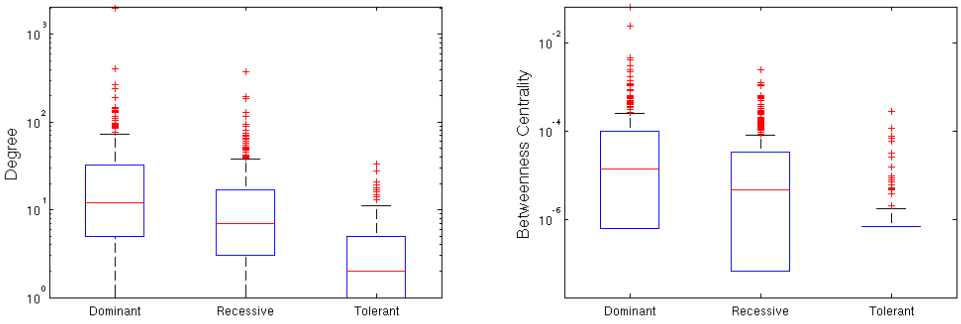


Figure 4.1: Degree and betweenness centrality in protein interaction networks. Differences between all three categories are significantly different (Wilcoxon ranked sum, $p < 10^{-6}$ for degree, $p < 10^{-4}$ for centrality)

### 4.3.2 Genetic Interaction Networks

Sampling biases are a well documented concern in PPI networks [76]. In 2006, known interactions were estimated to comprise only 10% of the full network [80]. While the number of physical interactions held in BioGRID has grown considerably since (from 26700 in December 2006 to 146800 in June 2014), the mapping is still likely to be incomplete. Well studied proteins are therefore likely to have a higher number of documented interactions. Even high-throughput methods which theoretically should sample the network randomly have been found to be biased towards evolutionarily conserved and high abundance proteins [239]. This may mean that known disease associated genes, which are likely to be

both well studied and evolutionarily conserved, have an artificially high number of interactions compared to the rest of the network.

To circumvent this bias, degree and centrality were also examined in a model of a human genetic interaction network. Traditional double-knockout interaction data is very sparse in human: for example, BioGRID holds only 1643 unique genetic interactions (June 2014) for *Homo Sapiens*. This has motivated attempts to assess human genetic interactions in other ways: Lin et al developed a method of re-appropriating data from radiation hybrid (RH) panels, a technique normally used for genetic mapping, to infer human genetic interactions [138]. This dataset has the advantage of being comprehensive (assessing 99% of possible gene pairs). However, the paradigm differs from the traditional double knock-out and has been less well studied.

In radiation hybridization experiments, a donor cell carrying a selectable marker is radiated, causing random DNA fragmentation. The irradiated donor cells are then fused with host cells lacking the marker. Fused cells are grown on selective media, leading to survival of host cells having incorporated the marker, along with a random set of other DNA fragments from the donor cell. The survival rate of clones is assumed to depend on which fragments of DNA are co-retained. Genetic interactions are therefore inferred from increased or decreased survival rates when two genes are co-retained (Figure 4.2).

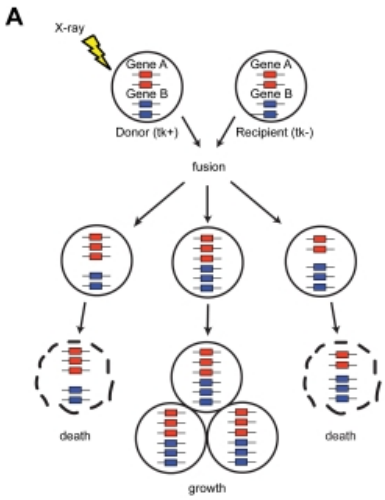


Figure 4.2: Inference of genetic interactions from radiation hybrid experiments. The figure illustrates a genetic interaction between gene A and B. The fused cell receiving DNA fragments containing only gene A or gene B from donor cell results in cell death. Receiving both gene A and gene B, however, results in survival, suggesting an interaction between the two genes. Figure modified from [138].

The RH genetic interaction network is therefore different from the standard double knock-out: instead of the effect of losing function in two genes, this inter-

action captures the effect of having an extra copy of two genes. The functional significance of these interactions has not been experimentally validated, thus interpretation of the RH network requires caution. Lin et al found that while the RH and PPI networks shared a number of global network properties, there was only limited overlap between the networks themselves [138]. This suggests the two approaches capture a different form of interaction, potentially making the RH network a valuable complementary approach.

The result observed in the PPI network was replicated in the RH network: dominant disease genes had the greatest degree and centrality and the LoF tolerant genes the lowest. All differences between the groups were significant (Wilcoxon ranked sum, $p < 0.005$ for both degree and centrality), but the effect was less pronounced than in the PPI network.

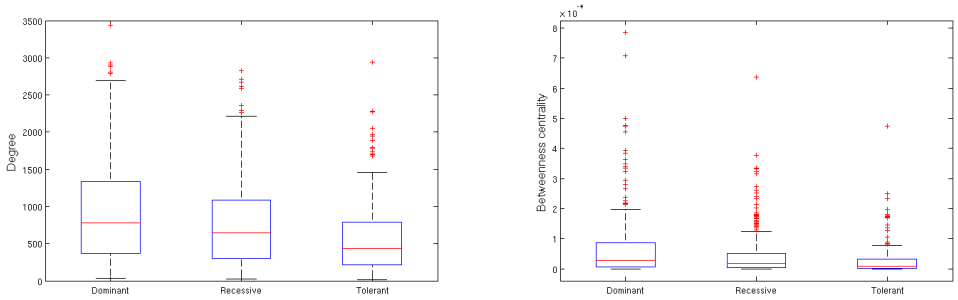


Figure 4.3: Degree and betweenness centrality in radiation hybrid genetic interaction networks. Differences between all three categories are significant (Wilcoxon ranked sum, $p < 0.005$ for both degree and centrality).

The greater difference in degree and centrality between the gene groups in the PPI network compared to the RH network is not necessary indicative of sampling biases playing a role in the PPI network: the change may simply reflect differences in the nature of the interactions captured by the two networks. The effect being observable in the RH network, however, does suggest that it is not attributable simply to sampling bias.

### 4.3.3 Metabolic Networks

As discussed previously, the relationship between gene essentiality and PPI network centrality is well documented. Whether a similar relationship exists in metabolic networks is not as clear. Khurana et al [111] find a negative correlation between gene significance and metabolic network degree and a positive correlation between metabolic network degree and the number of paralogues. The authors suggest that these paralogues may be involved in compensating for enzyme deactivation, thus making high degree genes more likely to be loss of function tolerant. Rio et al [199], however, find that while single measures of connectivity are not predictive of functional importance, combining a number of

centrality measures does predict essentiality - with higher centrality correlating with higher likelihood of essentiality.

The lack of consensus on the role of centrality in metabolic networks may be partly due to metabolic networks being conceptually less well defined than protein interaction networks. Metabolic networks are often represented as bipartite networks, with nodes representing either reactions or metabolites. Gene (or enzyme) networks are constructed from these bipartite networks by connecting enzymes which catalyse reactions involving the same metabolite (as illustrate in Figure 4.4).

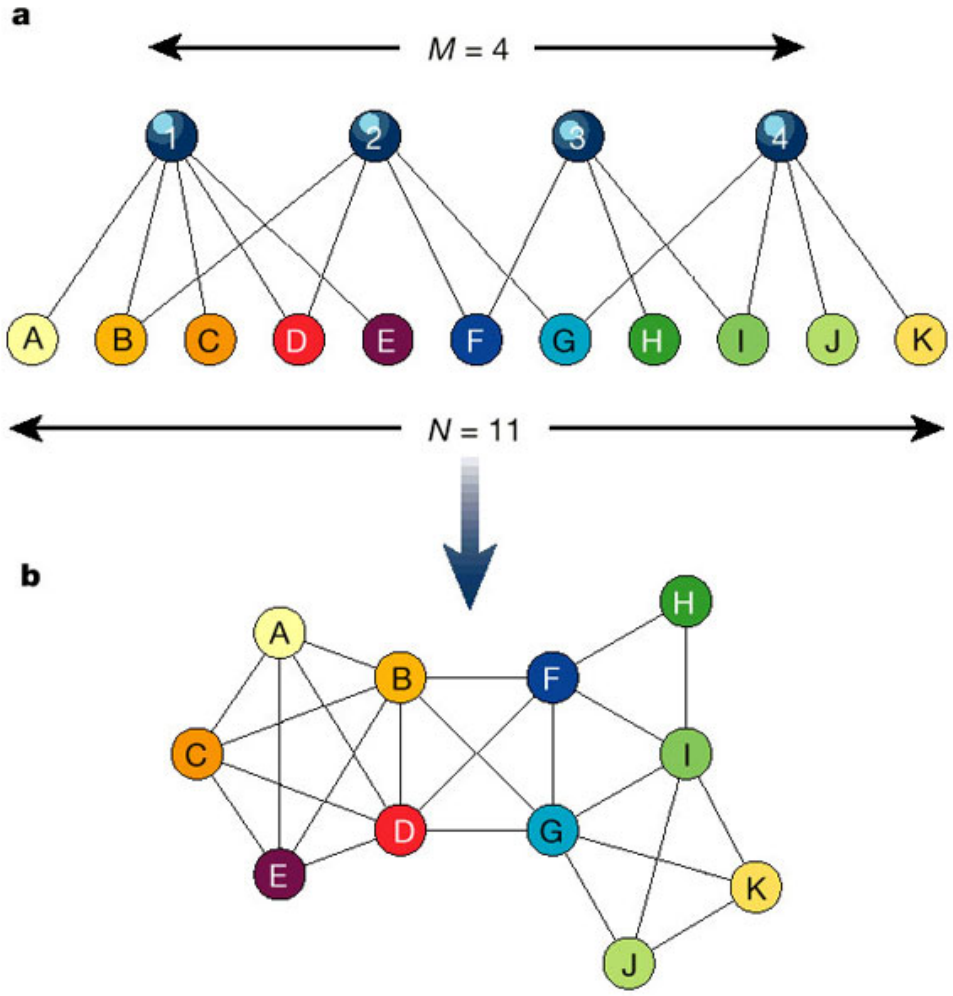


Figure 4.4: Construction of gene-gene (or enzyme-enzyme) metabolic networks from metabolite-enzyme networks: enzymes interacting with the same metabolites are connected, usually with edges weighted according to the number of shared metabolites. Figure from [219].

Particularities of this conversion process may vary, for instance in the treatment of edge directions (i.e whether a metabolite is created or consumed in a reaction) and the weighting of edges. Furthermore, many authors removed highly connected 'currency metabolites' such as ATP because these molecules

participate in a large number of functionally diverse reactions, thus diluting the functional information available from the network topology. Various approaches to currency metabolite have been proposed ranging from simply removing all metabolites with connectivity higher than a specific threshold [63] to heuristic methods removing highly connected metabolites until a specific network property (such as modularity [92]) is maximised. (The implicit assumption is that the chosen property correlates with how informative the network is.) Finally, in large scale metabolic networks, cellular compartmentalisation means that reactions appearing to share the same metabolite are in fact physically separated. Again, how this is treated in metabolic network generation differs between authors.

This ambiguity around the treatment of metabolic networks is apparent in work discussed above: Rio et al remove a total of 10 currency metabolites, while Khurana et al do not discuss currency metabolites. We were therefore interested in how metabolite removal affects how well functional importance is captured in these networks.

The human metabolic network was downloaded from Recon 2 [226]. This network was in the form of a reaction-metabolite bipartite network. Enzymes were mapped onto reactions they catalyse, after which an enzyme-enzyme network was generated as described above (discarding edge directions and weighting edges by the number of shared metabolites to give a weighted, non-directed network).

Without metabolite removal, no relationship was found between the essentiality and centrality in the metabolic network, differing from the negative correlation reported by Khurana et al [111]. To investigate the effect of metabolite removal, we adopted the approach laid out by Huss et al [92]: metabolites were removed in order of degree and the change in network modularity computed. However, unlike Huss et al, there was no clear cut-off point that optimised modularity. Therefore, the choice of which metabolite to remove was based on lists of currency metabolites in the literature [142,210,240], leading to removal of 23 distinct metabolites, corresponding to 149 nodes in the network as some metabolites were present in multiple compartments. However, even with metabolite removal, no significant relationship between essentiality and centrality was observed.

Based on these results, the metabolic network was excluded from further study.

## 4.4 Prediction Using Centrality

Our results suggest dominant disease, recessive disease and LoF tolerant genes have different centrality in PPI and GI networks, suggesting these networks may be used in building a classifier. There are various ways in which the centrality measures may be used to predict the category of a gene. Here we investigate

nearest neighbour approaches: a gene is classified according to the class of the k genes with a centrality most similar to its own. For simplicity, we include only degree, not betweenness centrality, in building the predictor.

Classification was benchmarked using leave-one-out cross-validation, on random subset of 100 genes from each gene class (to avoid bias due to different numbers of each type of gene - see Table 4.1). This random sampling was repeated 100 times - the average performance is shown in Figure 4.5 and Figure 4.6.

Both the PPI and GI classifiers outperform a random predictor (expected performance 0.33), with the PPI classifier having greater maximal performance. Interestingly, for both classifiers, the performance on the tolerant gene set increases with the number of neighbours included in the predictor, while the performance on the dominant set decreases. A possible explanation for this is the way the degrees in the two sets are distributed: the distribution is right skewed and the genes with exceptionally high degrees are more likely to be dominant than recessive (see Figure 4.1 and Figure 4.3). Thus, as the number of genes included in the predictor is increased, on average, the new genes in the neighbourhood of the gene to be classified are more likely to be tolerant than dominant. Thus, the proportion of genes classified as tolerant increases, while the proportion of those classified as dominant decreases.

## 4.5 Guilt-by-Association

Networks can also be used for guilt-by-association type prediction of protein function. As discussed above, previous work suggests that proximity in a PPI network to other haplosufficient or haploinsufficient genes is predictive of a gene's behaviour [90]. The commute-time kernel introduced in Chapter 2 allows a more sophisticated treatment of network proximity than approaches based on shortest path length only.

Here, we use the commute-time kernel matrix (built from the human functional association network from the STRING database [101]) for gene classification using a k-nearest neighbours approach: a gene is classified into the same category as the k genes with highest similarity to it. We also explore a weighted approach, where the similarities of the k nearest genes are summed together and the gene is classified according to this score.

As before, performance was benchmarked using leave-one-out cross-validation on a sample of 100 genes from each gene category (dominant, recessive and tolerant). The average performance is illustrated in Figure 4.7 and Figure 4.8. The weighted approach outperforms the unweighed approach, while both methods outperform the degree-based approaches. With the guilt-by-association method, performance is higher on the dominant genes and is improved with increasing k, while the performance on the tolerant genes deteriorates. This behaviour can be attributed to the higher degree and centrality of dominant genes: these

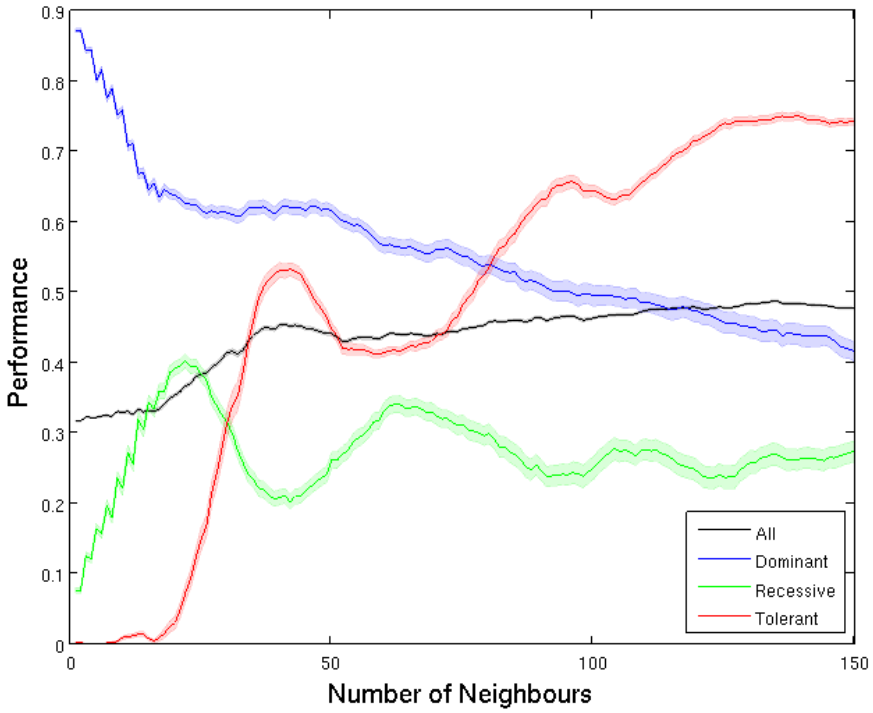


Figure 4.5: Performance of a nearest neighbour classifier for different values of k (nearest neighbours), using degree from PPI network. The figure shows percentage of correctly classified genes (overall and in each category), averaged over 100 random samples of 100 genes from each category. The shaded region represents the standard error of the mean.

genes will have higher than average similarity to all nodes in the network. Thus, as more genes are included in the prediction, the additional genes will have a higher probability of being dominant genes, thus increasing the probability of classifying genes as dominant. In the unweighed predictor, at around k=100, this leads to all tolerant genes being misclassified. The effect is less pronounced in the weighted model, because the effect of additional genes is smaller, as they have, by definition, lower similarity than the nearest neighbours.

## 4.6 Integrated Prediction

Our analysis thus far suggests centrality in the PPI and genetic networks and proximity in the functional association networks can be used to distinguish between dominant disease, recessive disease and loss of function tolerant genes. Next, we investigated whether these information sources could be combined to improve overall performance.

For simplicity, the analysis here is restricted to genes present in all three networks (PPI, GI and functional association from which the kernel is derived). Figure 4.9 shows the performance of individual data sources on this data set.

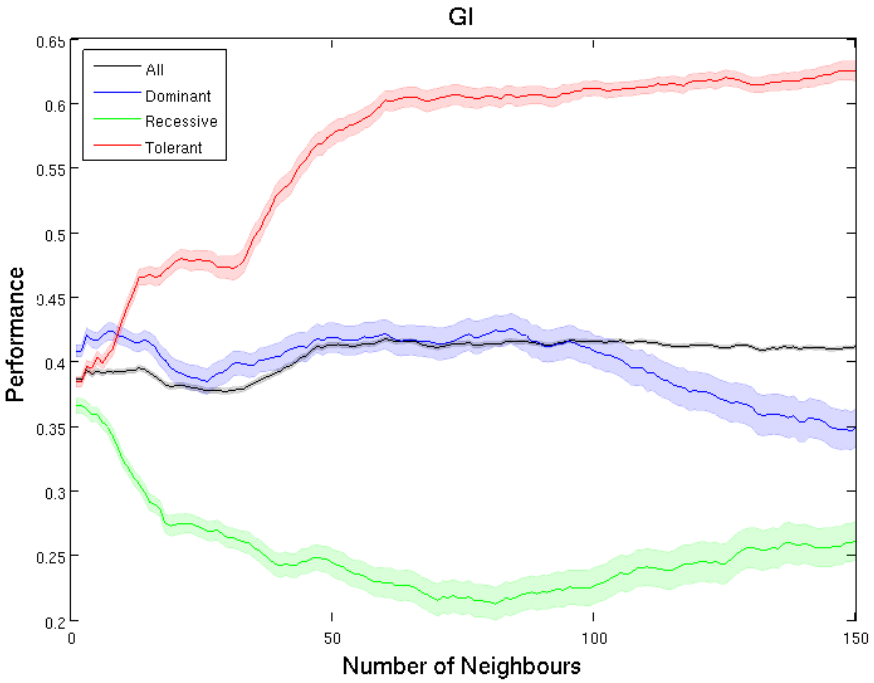


Figure 4.6: Performance of a nearest neighbour classifier for different values of k (nearest neighbours), using degree from GI network. The figure shows percentage of correctly classified genes (overall and in each category), averaged over 100 random samples of 100 genes from each category. The shaded region represents the standard error of the mean.

Overall, on this smaller dataset, the predictive performance is lower than on the full dataset, for all networks.

The predictor assigned each gene a score reflecting the likelihood of belonging to each category. This combined score is a linear combination of each data source's individual scores. For the GI and PPI networks, the score is the number of neighbours within the k nearest neighbours belonging to each category. For the kernel predictor, the score is the sum of the similarity scores of the genes in each category within the k nearest neighbours. Performance was benchmarked using leave-one-out cross validation on sets of 20 random genes from each category, averaged over 100 samples. The optimal number of neighbours for each information source was determined by further cross-validation within each fold.

Figure 4.10 shows the performance of the combined predictor at different relative weightings of the information sources. Optimal results are achieved through integration of the PPI data and the functional association kernel. The addition of GI data does not improve performance. This is not surprising, given the lower predictive performance of the GI data-based classifier (Figure 4.9). It remains possible, however, that a more sophisticated prediction approach would exploit the GI data more successfully.

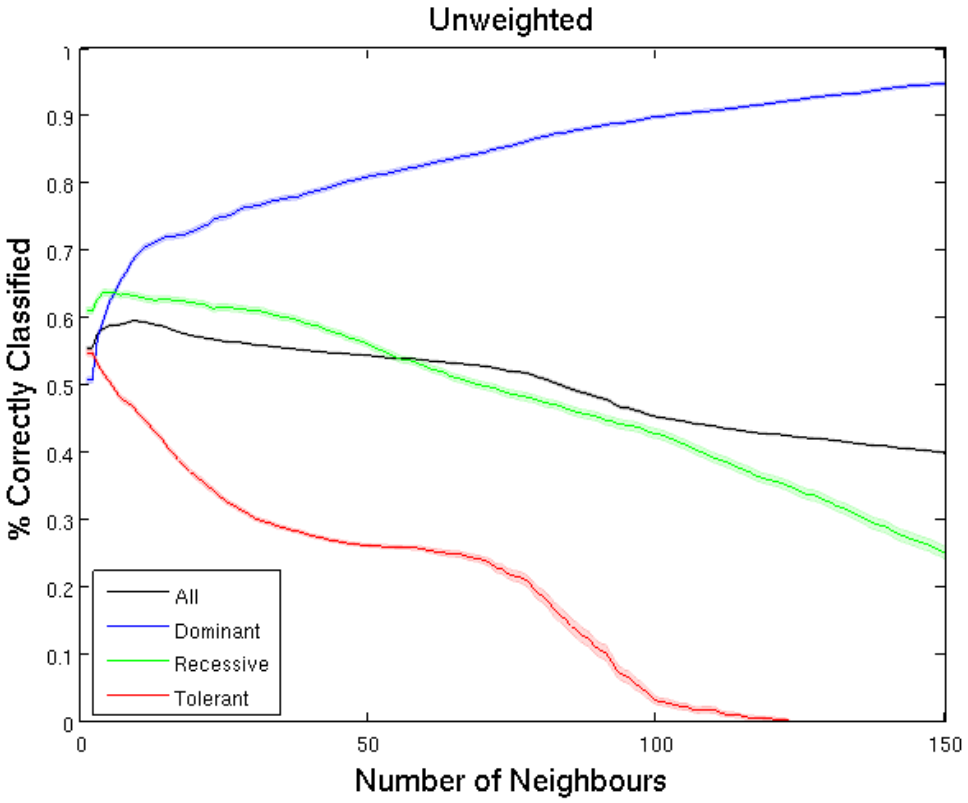


Figure 4.7: Performance of a kernel-based k nearest neighbour classifier for different values of k for an unweighted (gene classifed based on the number of genes in each category in its k nearest neighbours) classifier. The figure shows the percentage of correctly classified genes averaged over 100 random samples of 100 genes from each gene class (dominant, recessive and tolerant).

## 4.7 Discussion and Further Work

This section explores approaches for distinguishing between genes that are dominant disease causing when mutated, recessive disease causing when mutated and tolerant to loss of function.

Our results suggest that centrality and degree in PPI and GI networks as well as proximity in functional association networks can be used to discriminate between the three gene categories. Furthermore, we found that combining degree information from PPI networks and proximity information from functional association networks outperforms either predictor alone.

As discussed in Chapter 2, it has been suggested that the performance of guilt-by-association type predictors is dominated by gene degree [64]. It is interesting to note that in this work, inclusion of the guilt-by-association data improved performance over use of degree information alone. It remains possible, however, that the improvement comes from the inclusion of additional network data (i.e. the STRING functional association network the kernel was derived from) as opposed from the use of guilt-by-association specifically. It

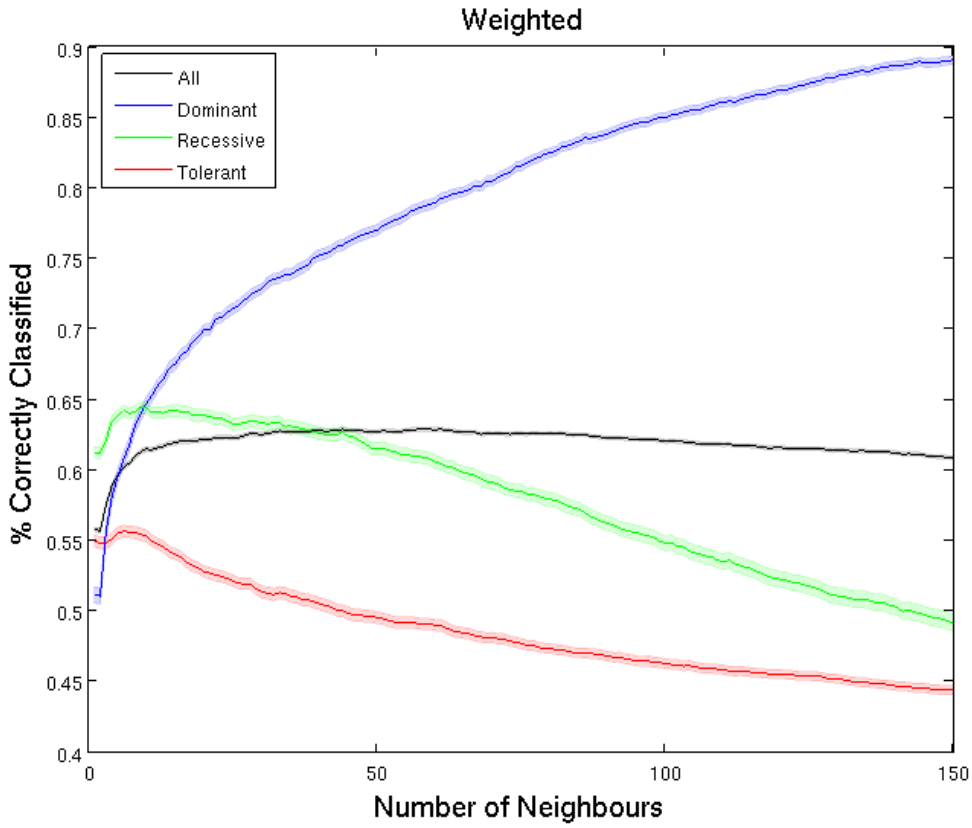


Figure 4.8: Performance of a kernel-based k nearest neighbour classifier for different values of k for a weighted (the similarities of the genes in each category in the k nearest neighbours are summed) classifier. The figure shows the percentage of correctly classified genes averaged over 100 random samples of 100 genes from each gene class (dominant, recessive and tolerant).

might therefore be interesting to control for this explicitly, by comparing the predictive performance of degree and guilt-by-association based predictors using the same network data.

Degree and centrality in metabolic networks were not found to be useful predictors. It is worth noting, however, that the number of genes present in the metabolic network was considerably smaller than in the other networks (see 4.1). It is therefore possible that the lack of predictive power in the metabolic networks was due to low coverage, rather than an inherent property of the network.

This section only explored relatively simple prediction algorithms. It may be interesting to investigate whether more sophisticated algorithms will further improve performance. Particularly, in this work, the addition of degree information from the GI network did not improve the performance of the predictor using PPI degree information and the kernel data. It would be interesting to explore whether a different prediction approach would make this dataset more useful.

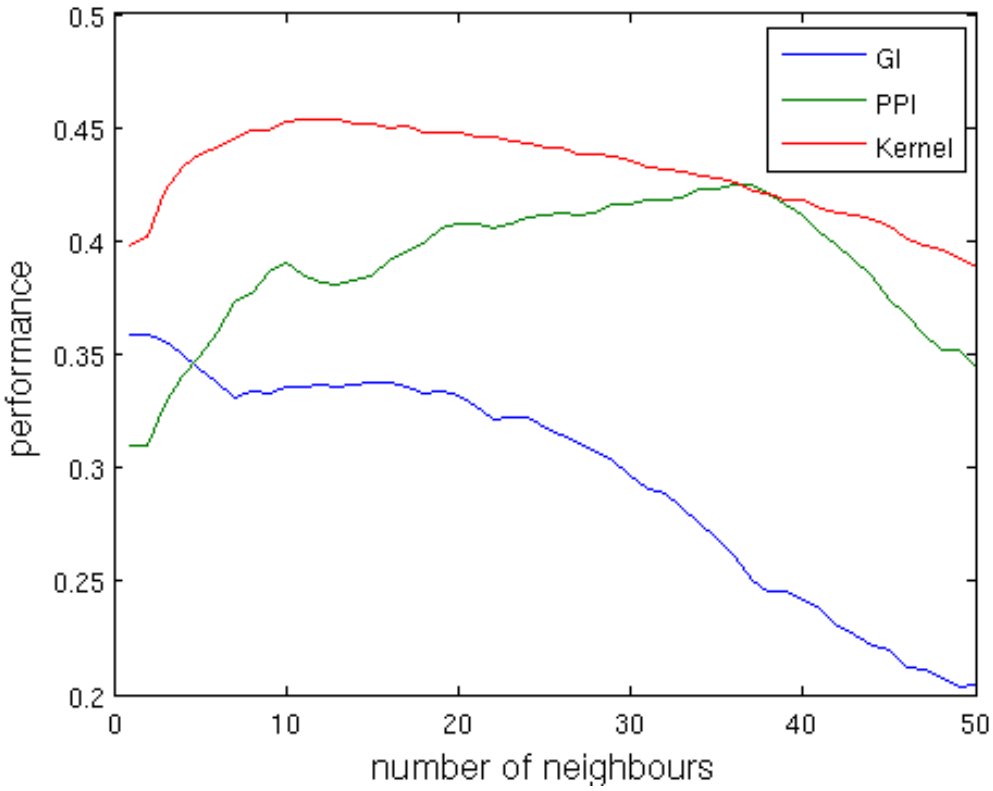


Figure 4.9: Performance of three data sources (GI network, PPI network and kernel) on the set of genes common to all three sources. The figure shows the proportion of correctly classified genes in random samples of 20 genes from each category, averaged over 500 draws.

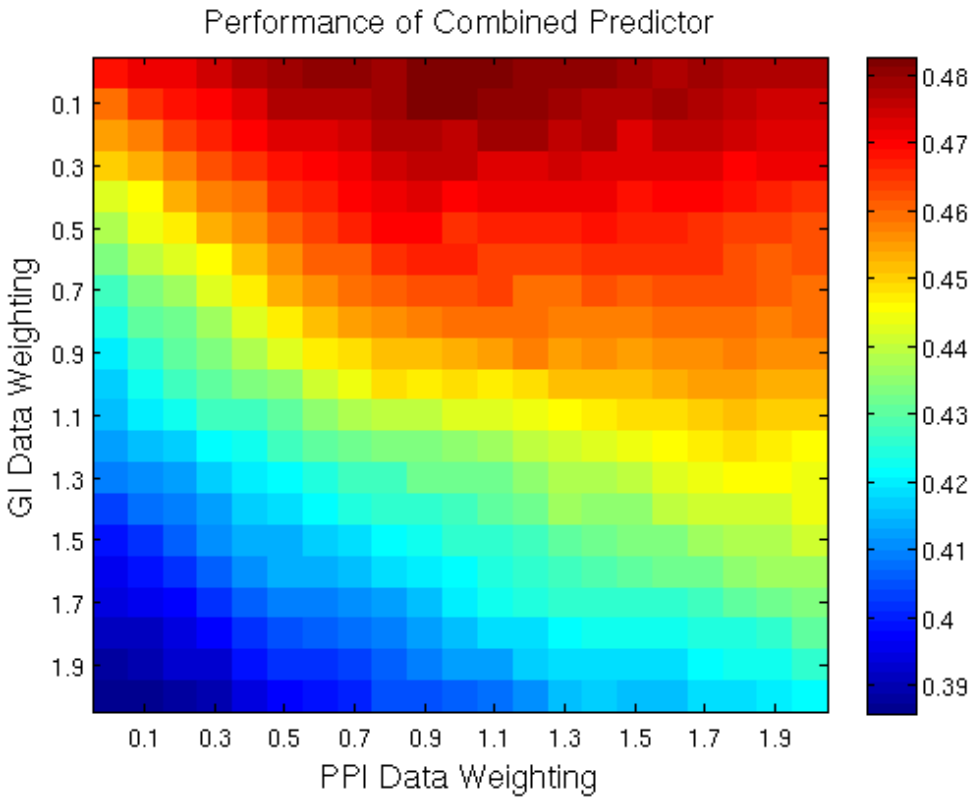


Figure 4.10: Performance of the combined predictor using the kernel, PPI and GI data, for various relative weightings of the different information sources. The kernel predictor always has a weight of 1 - the PPI and GI data are given the weighting indicated on the x and y axis respecively. The figure shows the proportion of correctly classified genes in random samples of 20 genes from each category, averaged over 100 draws.

# Chapter 5

# Network Approaches to Modelling the Stress Response in Fission Yeast

## 5.1 Introduction

### 5.1.1 Stress Response

The ability to maintain function in the face of external perturbations (environmental 'stress') is important for all organisms. It is particularly crucial for micro-organisms, such as yeast, which cannot relocate to escape these perturbations [140]. Micro-organisms adapt to environmental changes through rapid and significant rearrangement of their regulatory systems. This rearrangement, known as the stress response, is orchestrated at multiple levels of regulation (transcriptional, post-transcriptional, translational) [126].

Genome wide technologies have produced insight into global stress-induced changes in gene expression: in response to stress, cells shift resources away from metabolism, growth and proliferation, and towards protective mechanisms [140, 204]. This redirection of resources is seen in both budding and fission yeast, and in response to multiple stress types. In addition to this core response, however, other components of the stress response are fine-tuned to the type and strength of the specific stimulus [58].

While most of the changes in gene expression associated with the stress response are transient [140], exposure to stress has a lasting effect on the cell: the stress response results in higher tolerance against future insults of the same kind, as well as against other stressors [19]. This cross-protection effect is attributed to lasting activation and/or expression of stress proteins [125].

Stress induced genes have noisier expression than growth related genes and show higher variability between cells and conditions [132, 169]. Consequentially, variable environments lead to higher levels of heterogeneity within a population of cells, making it more likely for at least part of the population to survive

a change in conditions ('bet hedging') [140]. Additionally, these stress induced genes also show high inter-species variability, suggesting they evolve more rapidly (greater 'evolvability') [181, 227].

Thus, it appears that stress not only promotes short-term adaptation to allow maintenance of function, but contributes to long-term resilience and acts as a major driving force of evolutionary change [140].

### 5.1.2 Studying Changing Networks

Network approaches are a powerful tool in the analysis of genome wide data sets and a useful framework for capturing the global state of a system. Recently, there has been increasing interest in understanding how cellular networks differ under different conditions - for example cell type, disease or environmental perturbations.

Condition-specific networks are generally generated by combining a static PPI network with condition-specific data. This is often only an approximation of the true differences, as this approach cannot distinguish between network rewiring (condition-specific loss or gain of interactions) and changes in network state (such as, for example, changes in the expression levels of proteins) [94]. Despite this, these approximations of condition-specific networks are often the best way to probe system level changes, because condition-specific interaction mapping studies are still relatively rare.

There are two mains approaches to differential network analysis. One approach is to identify subnetworks that are only active under particular conditions [41, 95, 207]. For example, by integrating known transcriptional regulatory interactions with gene expression data to find condition-specific transcriptional regulatory networks, Luscombe et al. showed large scale topological difference between conditions [141]. Interestingly, these condition-specific networks could be classed, based on their structure, into two categories: endogenous (internal transcriptional programs) and exogenous (responses to external stimuli). Given that growth and proliferation fell into the former category, and stress response into the latter, this division bears resemblance to the two antagonistic gene expression programs discussed earlier.

Another example of such approach, applied specifically to stress, combined physical protein interaction, curated pathway, metabolic and gene expression data and revealed changes in local network topology in response to oxidative stress [71].

In an interesting extension of this type of approach, Komurov and White mapped the expression dynamics of proteins onto a protein interaction network [119]. This revealed two types of functional subnetwork: static modules composed of constantly expressed proteins and dynamic modules composed of proteins co-expressed in a condition dependent manner. Interestingly, given the link between the noisiness and evolvability of stress response gene expression

(see Section 5.1.1), both evolutionary rate and expression noise were higher in static module proteins. Furthermore, some of these static module proteins were found to be phenotypic enhancers of genetic mutations. The authors therefore suggest that fluctuations in the levels of these modules may thus contribute to bet hedging strategies through enhancement of cell-to-cell variability.

An alternative approach to working with condition-specific networks is to analyse changes to the network as a whole, instead of focusing on subnetworks. The advantage of this approach is that it enables detection of global changes in network structure. For example, Mihalik and Csermely generated distinct networks for stressed and unstressed states by weighting the budding yeast interactome by the abundance of the interacting proteins in each condition [152]. The authors reported a partial disassociation of this network under heat stress, with fewer connections between network modules. The authors suggest that this decoupling of modules represents a cellular survival strategy. The pruning of interactions could (i) increase network resilience to further damage by decreasing information flow between modules, thus minimizing the spread of damage [115]; (ii) represent the emergence of more specialized and autonomic functional units, which Mihalik and Csermely suggest could allow to cell greater behavioural flexibility [152]; or (iii), in networks where links have a metabolic cost, result from energy saving measures.

### 5.1.3 Work Undertaken

In the work presented here, we study stress induced changes to cellular networks, with a patricular focus on the modular structure of the network. Two complementary approaches approaches are used: co-expression and weighted protein interaction networks.

#### Co-Expression Networks

A co-expression network captures similarities in genes' patterns of expression. In these networks, nodes represent genes and edges connect genes with strongly correlated expression. Co-expression networks are interesting because strongly correlated expression suggests functional association [46]: proteins involved in the same function are likely to be co-regulated. Indeed, in functional interaction networks, such as STRING [101], co-expression is often one of the main components of the functional association score. Given the known biases of protein interaction networks and their low coverage, particularly in fission yeast, co-expression networks provide a distinct and complementary perspective.

Co-expression networks are generally constructed from gene expression data under different environmental perturbations. In this work, however, *genetic* perturbations were used instead of environmental ones, as outlined further below (see Section 5.2.1). This allowed the construction of co-expression networks for both stressed and non-stressed conditions.

An advantage of this approach is that it does not rely on combining condition specific data with a condition independent network, thus allowing the study of stress induced changes to the topology of the network. A potential limitation is that while high levels of co-expression are considered an indicator of functional association between genes, the extent to which changes in co-expression are indicative of changes in functional association has not been explicitly studied. However, it is reasonable to expect that proteins which interact under specific conditions would be more tightly co-regulated in these conditions. This idea is supported by well documented condition-specific changes in transcriptional regulatory networks [48, 141].

#### Protein Interaction Network

Protein-protein interaction (PPI) networks provide a complementary view of cellular state. As cellular function is carried out at the protein level, PPI networks have a more straight forward interpretation and may therefore potentially provide greater functional insight. For example, the idea of network rearrangement resulting from energy saving mechanisms is only relevant when interactions are associated with metabolic cost. Some protein interactions, such as phosphorylation, are indeed energy consuming.

To generate biologically meaningful condition-specific networks, the edges in the PPI network were weighted by the approximate probability of the interaction occurring in the stressed or unstressed state (estimated either by co-expression or the product of the protein abundances as discussed in Section 5.2.2). This differed from Mihalik and Csermely's approach, where edges were weighted by the *sum* of protein abundances [152]. This method was not used in this work as, although using the sum of protein abundances instead of the product has the advantage of giving less extreme changes in edge weights, the biological interpretation of this measure is unclear.

## 5.2 Methods

The different networks constructed and analysed in the Chapter are summarised in Table 5.1.

### 5.2.1 Co-Expression Network Construction

Gene co-expression networks were constructed using gene expression data from genetic variants, before and after exposed to oxidative stress (0.5 mM hydrogen peroxide, $H_2O_2$), as outlined in Figures 5.1 and 5.2. Spearman correlation coefficients were computed across the genetic variants for each gene pair, under both stressed and non-stressed conditions. To generate the networks, a specific number of gene pairs with the highest significant ($p<0.05$) correlation coefficients were considered connected, yielding an unweighted network. This approach was

| Network Name | Network Type | Dataset Used in Construction |
|---|---|---|
| Microarray | Co-Expression | Microarray expression data from multiple knock-out mutants |
| RNAseq | Co-Expression | RNAseq expression data from multiple genetically different fission yeast strains |
| Co-expression weighted PPI | PPI | PPI network from the iRefIndex database and the RNAseq expression data |
| Abundance weighted PPI | PPI | PPI network from the iRefIndex database and protein abundance data from Papadakis et al. (manscript in preparation) |
| Abundance weighted nitrogen starvation PPI | PPI | PPI network from the iRefIndex database and protein abundance data from Marguerat et al [147]. |

Table 5.1: Summary of the networks used in the analyses and the datasets used in their construction.

taken to ensure that stressed and non-stressed networks were of similar size. The robustness of the results was also verified by (i) including different numbers of edges in the network and (ii) thresholding at a specific correlation coefficient, instead of edge number. The effect of stress was found to be the same regardless of the method of network construction (see Table 5.2).

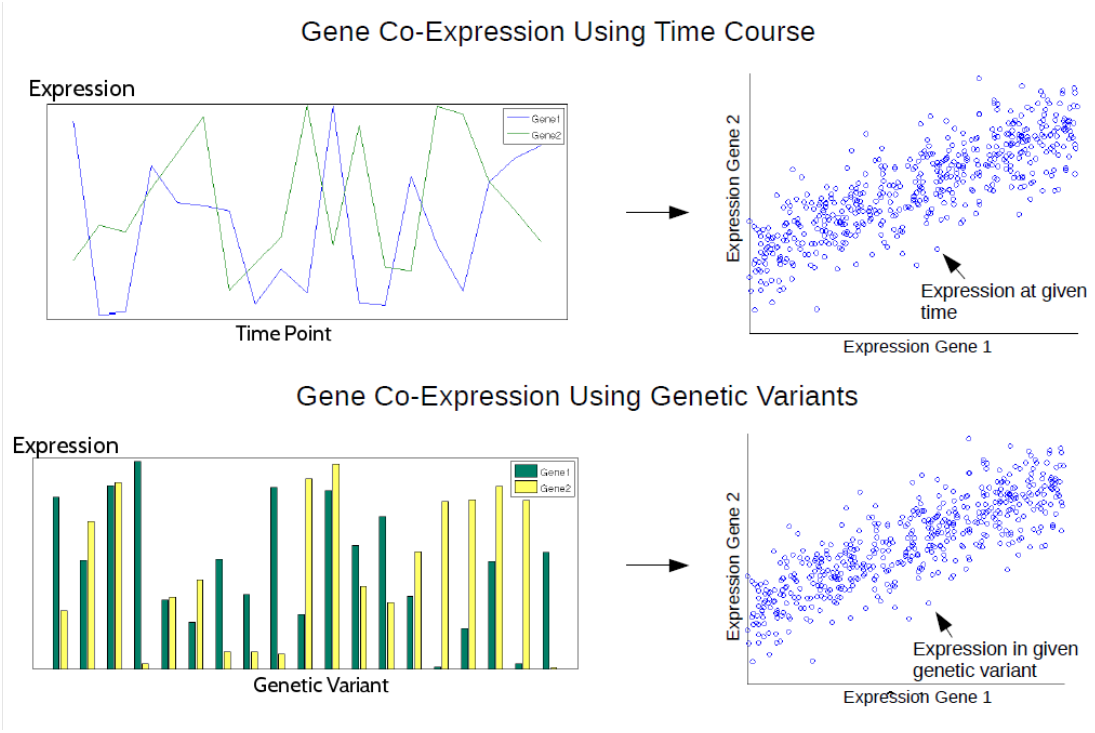


Figure 5.1: Outline of gene co-expression computation. Conventionally, the co-expression of two genes is calculated from the correlation of their expression time courses (top panel) or expression under different conditions. However, to generate distinct networks for stressed and non-stressed states, genetic variants (i.e. either knock-out mutants or genetic segregants) were used instead of a time course or environmental conditions.

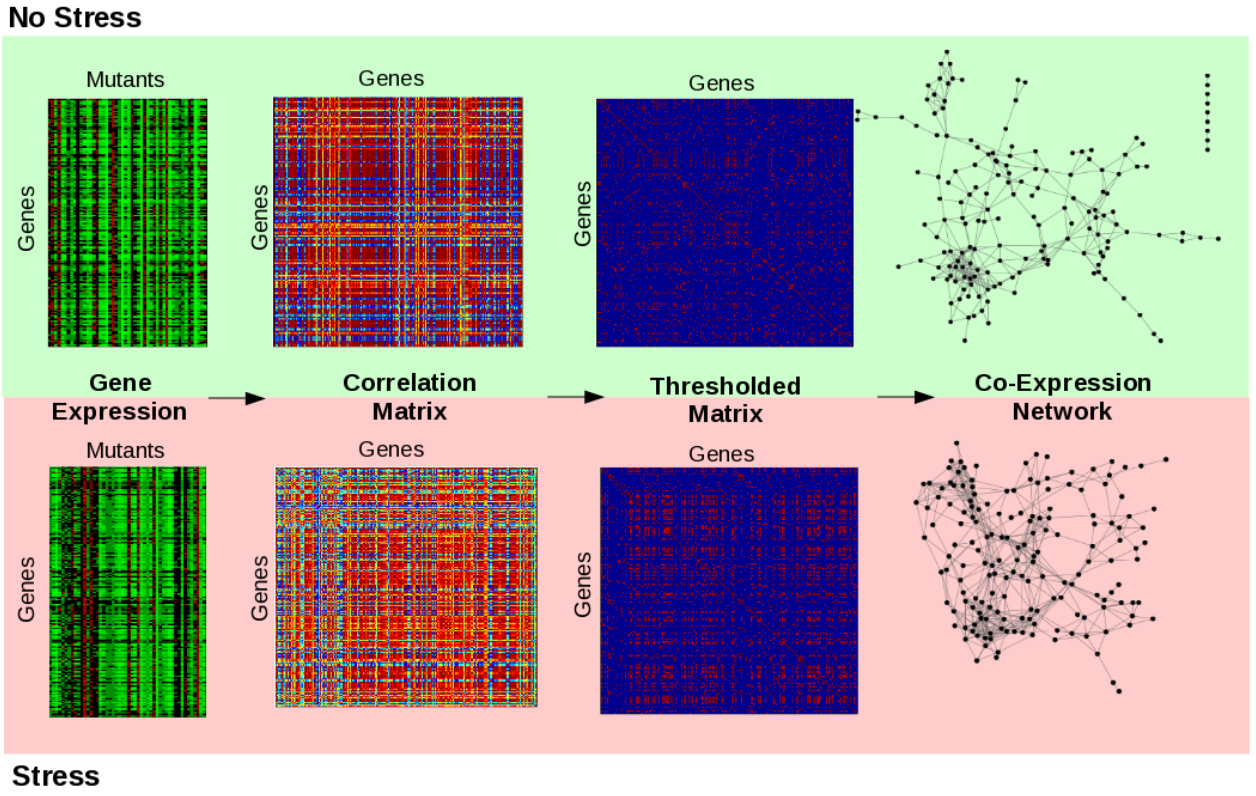


Figure 5.2: Outline of network construction. First, Spearman correlations in gene expression were calculated across different genetic variants for all gene pairs. These correlations were then thresholded to yield the adjacency matrix of the co-expression network. The genetic variants used in stressed and non-stressed conditions were the same, ensuring only the environmental condition differed between the two networks.

Two distinct sets of gene expression data were used to generate networks (giving a total of four networks: two non-stressed and two stressed). In the first data set, gene expression was measured using microarrays, while RNA sequencing (RNA-seq) was used in the second. The use of RNA-seq to quantify gene expression allows avoiding some of the problems associated with microarrays, such as cross-hybridization between highly related sequences and difficulties in accurately detecting low abundance species [243], resulting in higher replicability and detection of lowly expressed transcripts and alternative splice variants [200]. However, some studies suggest that co-expression networks generated from RNA-seq data may be less reliable than those from microarray data [68]. It is therefore useful to include both in the study.

Details of the two data sets are outline below.

### RNA-seq

These networks, henceforth referred to as RNA-seq co-expression, were constructed from gene expression levels measured by RNA sequencing in the Bähler laboratory from 117 genetic segregants (derived from crosses of genetically different wild isolates), at 0 and 60 minutes post exposure to 0.5mM hydrogen peroxide stress.

### Microarray Data

These networks, henceforth referred to as microarray co-expression, were built from gene expression levels in 8 knock-out mutants at 0 and 60 minutes after exposure to 0.5 mM hydrogen peroxide stress. The mutants used in the correlation calculation were atf31, ppr1, pap1, aft1/pap1, atf1, sty1 and pmk1. This was the only mutant data available for both stressed and non-stressed conditions. All expression data was collected in the same laboratory (Bähler laboratory), following the same protocol [169].

### Robustness of Microarray Correlations

Because the co-expression networks built from the microarray data set involved only seven mutants, they may be a less reliable measure of true correlations in gene expression. To verify, as far as possible, the robustness of the microarray network, each of the seven mutants was sequentially eliminated from the correlation calculation. For significant correlations above 0.9 this resulted in an average change of 0.02 in magnitude of the correlation coefficients. For significant correlations above 0.7, the change was 0.05. Generating networks from the recalculated correlations resulted in a 0.3% edge gain and 6.75% edge loss when thresholding at 0.9 (gain of 0.3% and loss of 2% when thresholding at 0.7).

As a further check, co-expression was re-computed using a wider pool of mutants including 24 additional mutants, which could not be used for network

construction, because they lacked expression data post exposure to stress. The co-expression, as calculated from the 7 mutants correlated (0.68 Spearman correlation coefficient) with the co-expression as calculated from the larger set of mutants. These results indicate that the correlation calculation is reasonably robust despite the relatively small number of mutants.

### 5.2.2 Protein Interaction Network Construction

The physical protein interaction network for S. pombe was downloaded from iRefIndex [194] a database consolidating interactions from a number of repositories (BIND [4], BioGRID [217], CORUM [203], DIP [205], HPRD [178], IntAct [109], MINT [136], MPact [74], MPPI [165] and OPHID [23]). To capture stress induced changes in the network, the interactions were weighted according to an approximation of the probability of their occurrence under specific conditions. As summarised in Figure 5.3, two distinct approaches were used in estimating the probability of interactions.

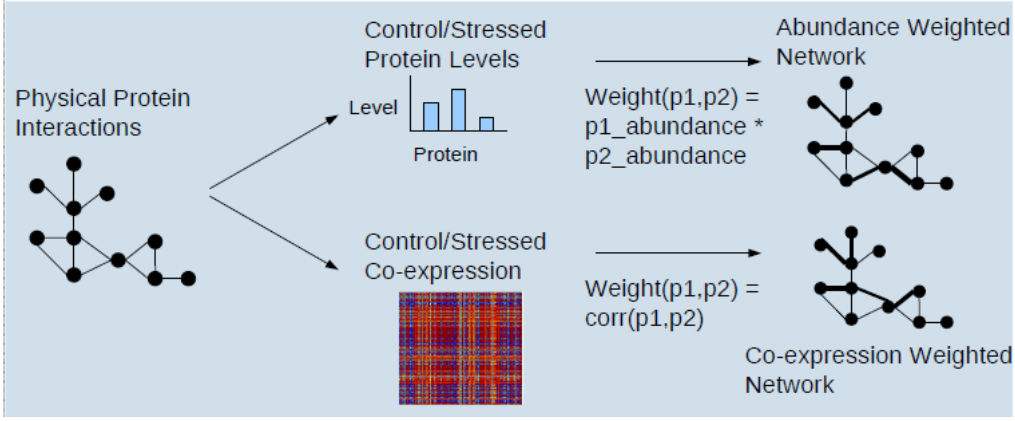


Figure 5.3: Weighted protein-protein interaction (PPI) networks were generated by condition-specific weighting of the physical interaction in fission yeast. The weight of the edge approximates the probability of the interaction occurring in the non-stressed or stressed cell. Two methods of edge weighting were used. 1) Abundance weighting, where the interaction between two proteins was weighted by the product of the proteins' abundances. To avoid bias against lowly expressed proteins, these products were normalized by the product in the non-stressed condition. 2) Co-expression weighting, where the interaction between two proteins was weighted by how correlated their expression is.

The first method was to weight the edge between two proteins by the product of their abundances. The protein abundance data used in this weighting scheme was collected by mass-spectrometry quantification of proteins from wild type fission yeast cells at 0, 60 and 240 minutes post exposure to 0.5mM hydrogen peroxide by Papadakis et al (manuscript in preparation).

The product of the protein abundances approximates the probability of the physical interaction occurring in the cell if we consider interactions to require the collision of randomly moving proteins. This idea is used, for example, in

mass action models of chemical reactions. A drawback of this approach is that interactions between highly expressed proteins will dominate heavily over interactions between lowly expressed proteins. Indeed, the multiplicative step may cause changes to lowly expressed proteins to be masked. Although there might be a weak correlation between the functional importance and expression level of a protein [166], interactions between lowly expressed proteins are not functionally insignificant. Therefore, in order to adjust for the bias against lowly expressed proteins, the approximated probability of interaction (i.e. the product of the abundances) was normalised by the approximated probability under non-stressed conditions. This normalised product was used to weight the interactions. The weights in the non-stressed network thus all become one, whereas the edge weights in the stressed network reflect the ratio of the probabilities of the interaction occurring pre- and post- stress.

The second way of weighting the interactions was to use the correlation coefficient (from the RNA-seq data set, as this represents correlation across a larger number of genetic variants, thus giving a better estimate of gene co-expression) as weights for the links. Negatively correlated protein pairs were assigned a weight of zero. This too is an approximation of the probability of the interaction occurring in the cell, as proteins both need to be present for the interaction to occur and the presence of the corresponding RNA can be a useful proxy.

#### Nitrogen Starvation

To investigate the network effects of a different form of stress, weighted abundance networks were also built from protein abundance data from proliferating and quiescent cells [147]. Quiescent cells had undergone 24 hours of nitrogen starvation prior to protein quantification. Full details of the quantification protocol can be found in Marguerat et al [147].

### 5.2.3 Network Modularity

Most graph partition algorithms divide networks into non-overlapping parts. However, modules in gene and protein networks are thought to correspond to functional units and that proteins may participate in multiple functions - there is therefore increasing interest in clustering gene and protein networks into *overlapping* modules (i.e. groups of nodes where nodes are permitted to belong to more than one group).

Various approaches have been proposed to perform this overlapping clustering. To ensure our results were independent of the particularities of specific module finding algorithms, two distinct methods of clustering were used:

1. Link Communities (LC)
2. ModuLand (ML)

### Link Communities

The Link Communities [2] algorithm clusters nodes into overlapping modules based on a non-overlapping clustering of the edges.

Edges are first clustered by computing a similarity measure, $S$, between edges. As illustrated in Figure 5.4, the similarity of two edges is based on the extent to which the nodes they connect share neighbours. Specifically, for edges $e_{i,k}$ and $e_{j,k}$, connecting nodes $i$ and $k$, and $j$ and $k$, respectively, S is given by:

$$S(e_{i,k}, e_{j,k}) = |n_{+(i)} \cap n_{+(j)})|/|n_{+(i)} \cup n_{+(j)}|$$

where $n_{+(i)}$ is the set of nodes $i$ and its neighbours. This type of measure (intersect divided by union) is known as the Jaccard Index. Edges are then assigned into modules by single-linkage hierarchical clustering. Finally, nodes inherit all module assignments of their edges, giving rise to overlapping network modules.

The similarity measure can be extended to weighted networks by re-expressing the Jaccard Index in terms of inner products. Specifically, if $A$ is the weighted adjacency matrix of the network, such that $A(i,j) = w(i,j)$, and $\mathbf{a_i}$ is row or column vector from this matrix, such that $\mathbf{a_i} = A(i,:) = A(:,i)$, $S$ can be expressed as:

$$S(e_{i,k}, e_{j,k}) = \mathbf{a_i} \cdot \mathbf{a_j} / (\mathbf{a_i} \cdot \mathbf{a_i} + \mathbf{a_j} \cdot \mathbf{a_j} - \mathbf{a_i} \cdot \mathbf{a_j})$$

Like the unweighted measure, the weighted measure captures the proportion of the nodes in the neighbourhood that are neighbours to both nodes $i$ and $j$, but gives connections with high weights greater impact.

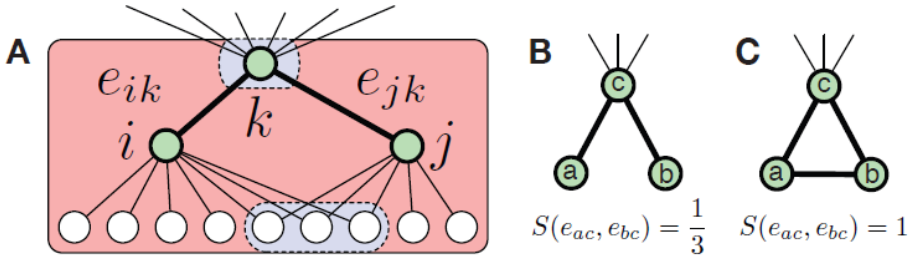


Figure 5.4: Example of how similarity measure $S(e_{i,k}, e_{j,k})$ is computed. (A) Similarity between edges $e_{i,k}$ and $e_{j,k}$, both connected to node $k$. The total number of nodes in the neighbourhood ($|n_{+(i)} \cup n_{+(j)}|$) is 12, while the number of shared nodes ($|n_{+(i)} \cap n_{+(j)})|$) is 4. Therefore $S = 4/12 = 3$. Two simple cases are illustrated in (B) and (C). Figure reproduced from reference [2].

In this work, a distance cut-off of 0.4 was used during the hierarchical clustering of the network edges, though stress induced effects on network overlap were found to be conserved using other (0.3-0.5) cut-off values. The effect of cut-off modification is discussed further in Section 5.3.2. For unweighted networks, the algorithm was implemented using a python script provided by Ahn et al [2]. For

weighted networks, the weighted version of the algorithm was implemented with custom written code in MATLAB. For unconnected networks, where paths do not exist between all pairs of nodes, only the largest connected component was considered.

### ModuLand

The ModuLand [122] family of algorithms compute overlapping modules by treating network modularity as a landscape, where small hills can exist as part of larger mountains, thus giving rise to overlapping module assignments, as illustrated in Figure 5.5. The 'elevation' is *community centrality*, a measure capturing the influence of nodes or edges on the rest of the network based on a perturbation-flow type calculation.

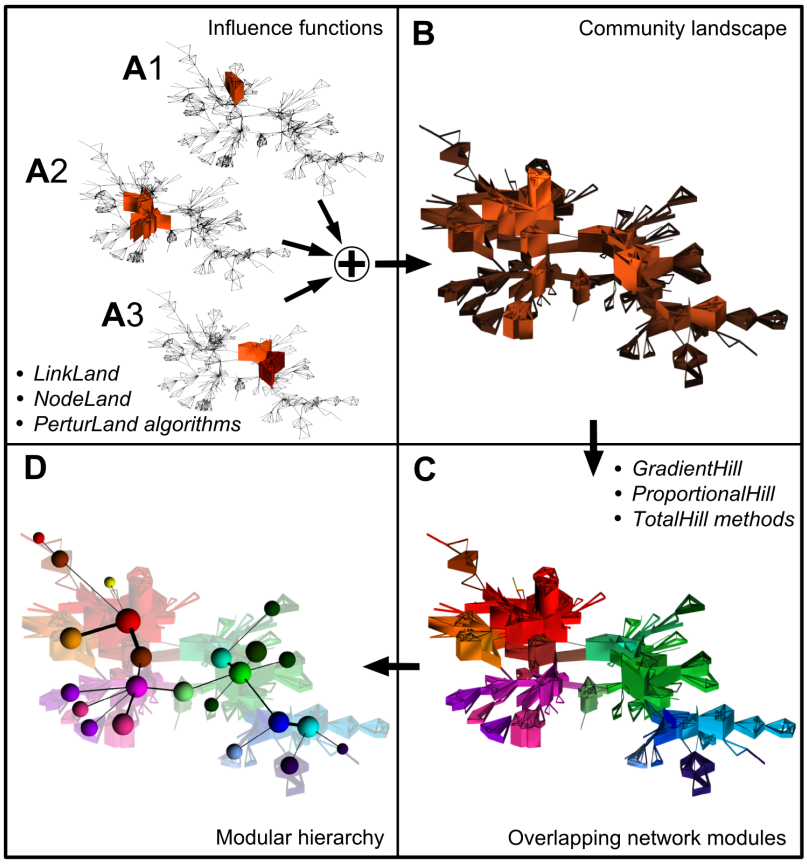


Figure 5.5: Summary of ModuLand module finding algorithm. (A) First, an influence function $f_s(i,j)$ is calculated for all nodes $s$. (B) These functions are then added together to give community centrality values for all edges. This is the 'community landscape' (C) Overlapping modules are found by finding local maxima in the community landscape (D) Finally, the modules themselves can be treated as nodes, giving rise to a higher level network. Figure reproduced from reference [122].

Briefly, for every node $n$, Moduland first determines a set of nodes $S_n$ with

a 'strong influence' on the node. The set is defined iteratively by starting with node $s$ and then adding the neighbouring node which maximises the density $d$ of the set, given by $d = \frac{\sum_{(i,j)_n} w_{ij}}{|S_n|}$. The set is thus expanded, until addition of further nodes no longer increase the density value.

An influence of function for node $n$ is then computed as $f_n(i,j) = w(i,j)$ if $(i,j) \in S_n$, and zero otherwise. The community centrality of the edge between nodes $i$ and $j$ is the sum of these influence functions from all nodes:

$$c(i,j) = \sum_s f_s(i,j)$$

Edges with higher community centrality than their neighbours (i.e local maxima) are assigned to individual modules (forming the module core), while the other edges are assigned to multiple modules proportionally to the centrality community values of their neighbours (referred to as the ProportionalHill module assignment method [223]).

ModuLand analysis was implemented using the ModuLand Cytoscape plugin.

## 5.3 Stress Induced Changes to Network Structure

### 5.3.1 Co-Expression Networks

The change in network structure is visualized in Figure 5.6.

#### Degree Distribution

The degree distribution, that is, the frequency distribution of the number of neighbours each node has, of a network can convey a lot of information about network structure, though by itself, it is not enough to fully characterize the network. Figure 5.7 shows the degree distributions of the microarray and RNAseq networks before and after exposure to stress. The microarray degree distributions are unusual in that they peak at a relatively high degree. Interestingly, the average degree of the different networks is roughly comparable (see table 5.2), suggesting the difference in degree distribution is not a trivial consequence of higher connectivity in the microarray network, but instead reflects a genuine difference in network structure.

Despite the difference in the shape of the degree distributions, inspection of the distributions suggest stress has the same effect on both: the distributions appear more uniform after exposure to stress. To assess this change quantitatively, the entropy, $H$, of the degree distributions $P(k)$ was calculated:

$$H = -\sum_{k=1}^{n} P(k) log(P(k)).$$

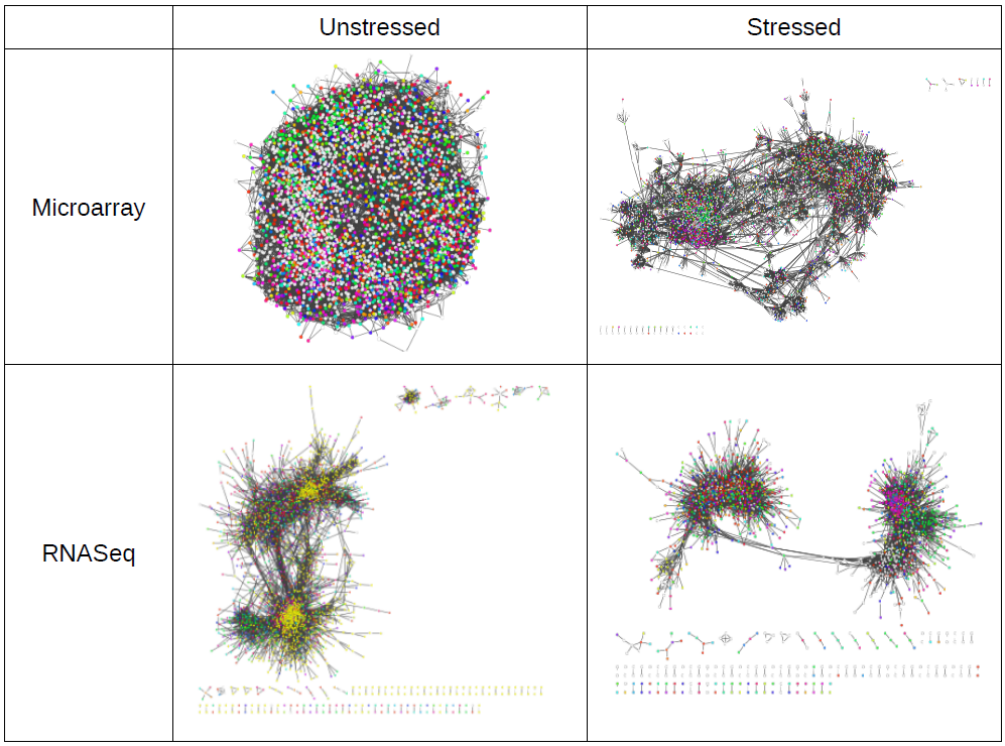


Figure 5.6: Visualization of co-expression networks before and after exposure to peroxide stress (0.5mM), showing the re-structuring of the network into more distinct modules. Nodes represent genes while the links between them represent a high level of co-regulation (that is, a high correlation in gene expression across genetic variants). The networks represented in this image have been thresholded at a specific number of edges (see Section 5.2.1). The stressed and non-stressed networks therefore have the same number of edges, but not the same number of nodes (for details on network properties, refer to Table 5.2). The visualizations were generated using force directed layout in cytoscape and nodes are colour coded according to GO category. Yellow nodes in the RNA-seq unstressed network are either non-coding RNAs or neighbours of a non-coding RNA.

Entropy captures the uniformity of a probability distribution. The greater the entropy, the more uniform the distribution. Interestingly, stress was indeed found to increase the entropy of the degree distribution in the RNA-seq networks, from 3.82 to 4.16 for networks thresholded at a specific edge number, and from 2.72 to 4.32 for networks thresholded at a specific correlation. In micorarray networks, however, stress decreased the entropy, from 4.16 to 4.02 for edge number cut-off, and from 4.41 to 4.17 for correlation cut-off).

It is unclear whether these differences reflect genuine differences in the behaviour of the interaction captured by RNA-seq and microarray co-expression. The difference in behaviour of the two networks may be due to biases in the microarray data resulting from the limited number of mutants the co-expression is calculated from. This is discussed further in Section 5.3.1.

For scale-free networks, it can be shown that maximizing the entropy of the degree distribution maximizes the network's robustness to node removal [241].

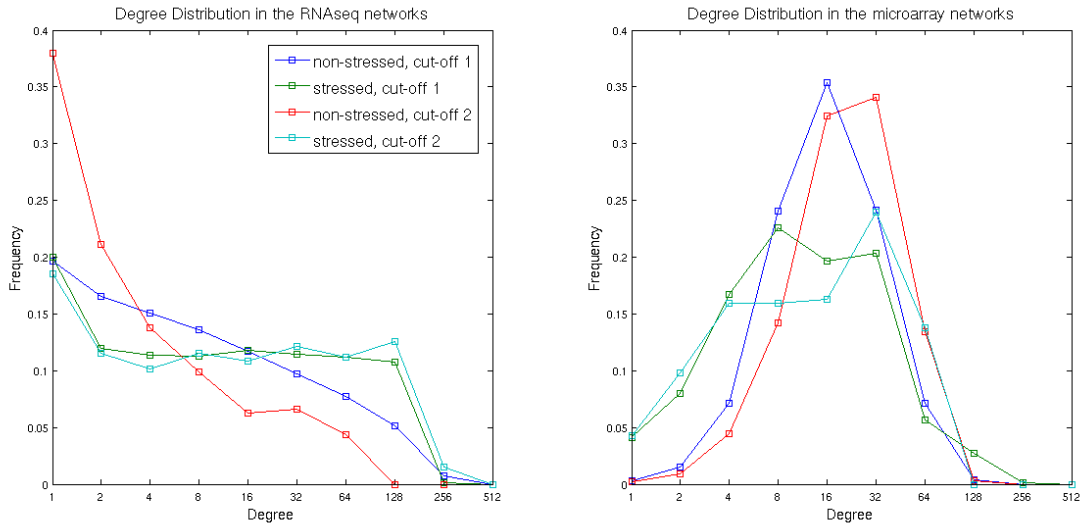


Figure 5.7: Degree distributions of the RNAseq and microarray networks using both fixed edge number (40 000 edges for RNAseq and 60 000 edges for microarray; referred to as cut-off 1 in the legend) and fixed correlation level (0.8 for RNAseq and 0.9 for microarray, referred to as cut-off 2 in the figure legend) to generate the networks. Note the logarithmic scaling of the x axis.

Although the distributions observed in the RNA-seq networks are not strictly scale-free, the change in degree distribution suggests increased resistance to node removal in the RNA-seq networks. This idea will be discussed further in Chapter 6.

### Network Statistics

In order to further quantify the change in network structure, various network statistics were computed. These measures do not necessarily have direct biological significance in themselves: however, they are a necessary starting point in understanding the changes to the network structure.

| Network | Time point (min) | Threshold | Nodes | Edges | Average Degree | Density | Average Shortest Path Length | Actual /Expected | Size of largest Component | Transitivity |
|---|---|---|---|---|---|---|---|---|---|---|
| RNAseq | 0 | Top 40000 | 2836 | 40000 | 28.21 | 0.005 | 4.85 | 1.87 | 2582 | 0.52 |
| RNAseq | 60 | Top 40000 | 1980 | 40000 | 40.4 | 0.010 | 6.33 | 2.71 | 1768 | 0.6 |
| Microarray | 0 | Top 60000 | 4240 | 60000 | 28.3 | 0.003 | 4.58 | 1.62 | 4240 | 0.38 |
| Microarray | 60 | Top 60000 | 4268 | 60000 | 28.12 | 0.003 | 6.10 | 2.15 | 4213 | 0.42 |
| Microarray | 0 | 0.9 | 4241 | 81946 | 38.64 | 0.0091 | 4.43 | 1.68 | 4241 | 0.43 |
| Microarray | 15 | 0.9 | 4359 | 91406 | 41.94 | 0.0096 | 4.45 | 1.71 | 4359 | 0.49 |
| Microarray | 60 | 0.9 | 4351 | 186982 | 85.95 | 0.0198 | 4.58 | 1.95 | 4351 | 0.57 |
| Microarray | 0 | 0.7 | 4242 | 309284 | 145.82 | 0.0344 | 3.20 | 1.56 | 4242 | 0.49 |
| Microarray | 15 | 0.7 | 4395 | 345603 | 157.27 | 0.0358 | 3.19 | 1.56 | 4395 | 0.53 |
| Microarray | 60 | 0.7 | 4356 | 619219 | 284.31 | 0.0653 | 3.25 | 1.63 | 4356 | 0.6 |
| RNAseq | 0 | 0.8 | 832 | 4304 | 10.35 | 0.0125 | 2.49 | 1.12 | 259 | 0.54 |
| RNAseq | 60 | 0.8 | 2273 | 54240 | 47.73 | 0.0210 | 8.66 | 1.85 | 2075 | 0.6 |

Table 5.2: Properties of co-expression networks at various time points during a peroxide stress (0.5mM) time course. Correlations in gene expression were calculated from expression data collected at specific points during the time course in different genetic variants. Networks were generated from the correlation data using two methods: either by drawing connections between a specific number of gene pairs with the highest correlations, or by considering gene pairs with a correlation above a specific threshold to be connected. As shown in the table, stress increases network density, shortest average path length and transitivity.

First, the average shortest path length was computed. This is the average minimum number of steps from one node to another in the network. The measure captures information about the network's connectivity structure. If the network is not fully connected (paths do not exist between all nodes), only the largest connected component is considered. In both microarray and RNA-seq data sets, stress was found to increase this measure (from 4.58 to 6.10 and from 4.85 to 6.33, respectively). This increase was conserved using different correlation cut-offs for network generation (see Table 5.2).

To assess the significance of this change, the average shortest path length for random permutations of the network were generated. In these permutations, the degree structure of the original network was conserved, but the edges were randomly re-shuffled. These permutations are referred to as degree preserving null models. Calculating the average shortest path length in these null networks gives the *expected* distribution of average path length for a network with the specific degree structure. In this work, 20 permutations were generated for each network of interest. This was deemed to be a sufficient number of networks because the variance of the average shortest path length ('expected' path length) of the 20 control networks was low. For each degree structure, the mean average shortest path length of the permuted networks was of the order of 2, while the standard deviation of the measure ranged from 0.001 to 0.004. Given the high computational cost of generating permutations and computing the average shortest path length of large networks, additional permutations were deemed unnecessary.

In both microarray and RNA-seq networks, stress was found to increase the actual average shortest path length significantly more than the expected average shortest path length ($p< 10^{-9}$, two-tailed t-test). This change in average shortest path length thus indicates a stress induced change in the structure of the network which is not simply explained by a change in the degree distribution.

The increase in average shortest path length is particularly noteworthy, given the stress induced increase in network density of the largest component (from 0.0067 to 0.0068 for the microarray and from 0.012 to 0.026 for the RNA-seq networks). Network density is the number of existing connections divided by the maximum possible number of connections for a fully connected network: a higher network density would thus be expected to yield a shorter path length, as more connections exist in the network. The increase in both path length and density suggests that stress leads to a restructuring of the network where links between 'local' genes (i.e. gene pairs that already have short paths between them) are increased, but connections to more 'distant' genes become fewer. In other words, the network becomes more modularised.

This idea was corroborated by a stress induced increase in transitivity, the probability with which two neighbours of a gene are also connected in the network. Stress was found to increase transitivity in both microarray and RNA-seq

networks (from 0.38 to 0.42 and 0.52 to 0.60, respectively).

Thus, the increase in path length, transitivity and density all suggest that stress creates a network structure with more tightly co-regulated modules, but fewer inter-modular connections.

### Modular Overlap

In this Section, we investigate the stress induced changes in the networks by looking at modular structure.

The modular structure of gene and protein networks is interesting because clusters of densely connected nodes are thought to correspond to functional units [14]. In general terms, genes and proteins are often considered to participate in more than one function. Consequently, there is increasing interest in clustering biological networks into *overlapping* modules - thus allowing nodes to belong to multiple network modules. This approach was used to further investigate the stress induced changes to co-expression network structure, specifically looking at module overlap. In the context of these networks, module overlap reflects the extent to which a single protein belongs to more than one set of tightly co-regulated proteins.

As seen in Figure 5.8, overlap decreases significantly in response to stress in both microarray and RNA-seq networks (Wilcoxon ranked sum test, $p< 10^{-6}$). This finding is robust when using different thresholds for edge inclusion (see Table 5.3). These results confirm the breakdown of the network into modules that have less interconnections between them.

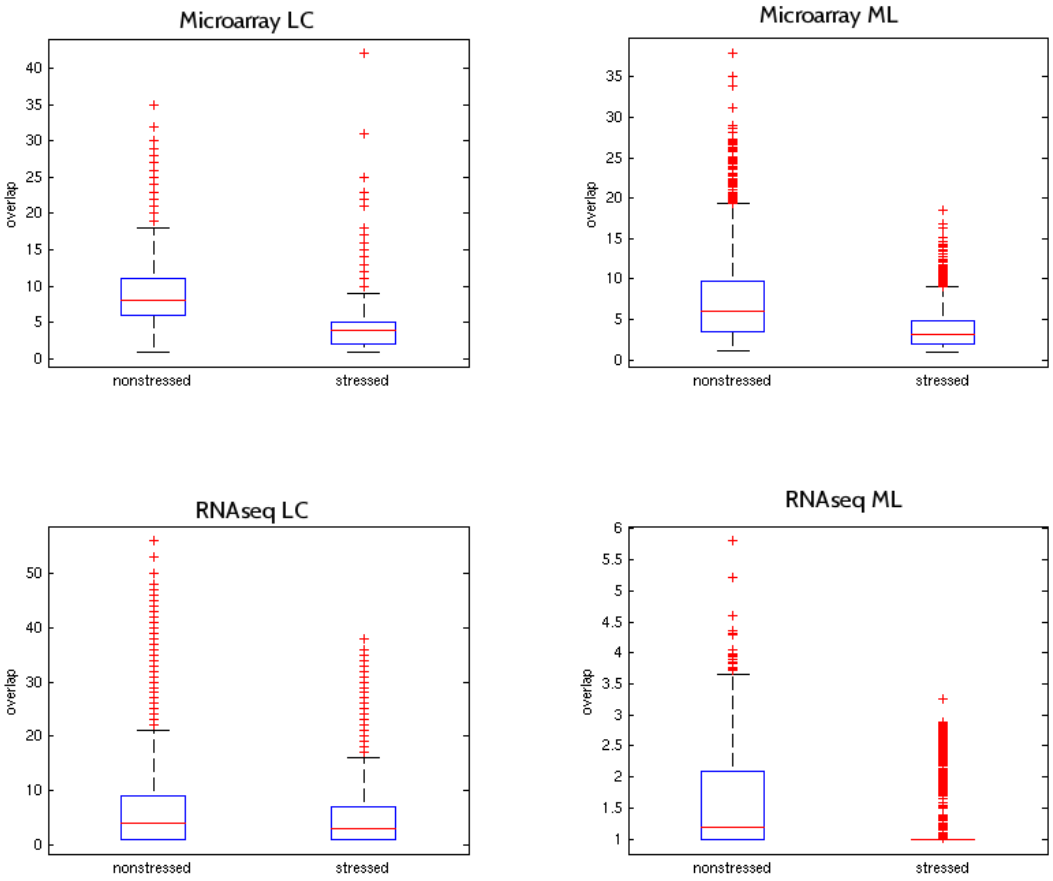


Figure 5.8: Changes to modular overlap in co-expression networks in response to oxidative stress (0.5mM hydrogen peroxide). The distinct module finding algorithms were used: ModuLand (ML) and Link Communities (LC, using clustering cut-off of 0.4, see Methods). For ModuLand modules, overlap was measured as ML overlap (see Methods), while for LC modules, overlap was measured as the number of modules a protein belonged to. Average LC overlap decreased from 8.88 to 3.43 for the microarray network and from 9.98 to 3.31 for the RNAseq network. Average ML overlap decreased from 7.15 to 3.63 for the microarray network and from 1.58 to 1.18 for the RNAseq network. All changes were significant (Wilcoxon ranked sum test, $p < 10^{-6}$).

| **Dataset** | **Time Point (minutes)** | **Threshold** | **ML Modules** | **Links between modules (ML)** | **Module Density (ML)** | **Average node-wise overlap (ML)** | **Average modules per node (link communities – S = 0.4)** |
|---|---|---|---|---|---|---|---|
| Microarray | 0 | 0.9 | 636 | 200981 | *1.00* | 94.78 | 7.60 |
| Microarray | 15 | 0.9 | 641 | 202952 | *0.99* | 92.36 | 7.70 |
| Microarray | 60 | 0.9 | 439 | 93144 | *0.97* | 42.81 | 5.74 |
| Microarray | 0 | Top 60000 | 458 | 102313 | *0.98* | *7.15* | 8.88 |
| Microarray | 15 | Top 60000 | 551 | 143856 | *0.95* | 7.06 | 6.50 |
| Microarray | 60 | Top 60000 | 339 | 27816 | *0.49* | 3.63 | 3.44 |
| RNAseq | 0 | Top 10000 | 131 | 39 | *0.00* | 1.10 | 3.67 |
| RNAseq | 60 | Top 10000 | 113 | 13 | *0.00* | 1.03 | 3.34 |
| RNAseq | 0 | Top 20000 | 152 | 228 | *0.02* | 1.47 | 4.63 |
| RNAseq | 60 | Top 20000 | 104 | 26 | *0.00* | 1.06 | 4.55 |
| RNAseq | 0 | Top 40000 | 131 | 252 | *0.03* | 1.58 | 9.98 |
| RNAseq | 60 | Top 40000 | 110 | 35 | *0.01* | 1.18 | 6.04 |

Table 5.3: Modular properties of co-expression networks, at various points during peroxide (0.5mM) stress time course, using two different methods of module detection: ModuLand (ML) and Link Communities (LC). Using both algorithms, modular overlap decreases during stress.

### Differences Between Microarray and RNA-seq Networks

The two types of co-expression network were both generated by analysing correlations in gene expression across different genetic variants. The genetic variants in the RNA-seq data are derived from crosses of genetically different wild isolates, and are therefore unlikely to show any specific biases. In the microarray data, on the other hand, all mutants were knock-outs of single genes with known regulatory functions in the stress response. This raises the possibility that the RNA-seq and microarray networks do not capture the same type of interaction and makes the interpretation of the microarray network more difficult.

In order to investigate this effect further, we tested the correlation between a gene's co-expression pattern as computed from the two data sets. The average correlation was 0.093 (range: -0.31 to 0.43 Spearman rank correlation). The low correlation between the two data sets suggests that there is a difference in the information captured by the networks. One explanation for this discrepancy could be that seven genetic conditions are not sufficient to accurately capture gene co-expression. However, as discussed previously, the calculation of the correlation from the microarray data was sufficiently robust to produce a fairly representative approximation of co-expression. A second possible explanation is a bias introduced because all mutants in the microarray data set are stress related. This could affect the co-expression network in two ways. First, the variability between the genetic conditions is low, explaining the higher average correlation in the microarray data set. This gives us less power to probe co-expression, meaning some patterns of co-regulation may therefore be missed. Second, all perturbation being stress related may confound the co-expression values for stress related genes: the expression of these genes may be dominated by the direct effects of the perturbation, masking effects of co-regulation.

Despite these points and the difference seen in degree distribution, the stress-induced changes in modularity are remarkably consistent in the two networks, suggesting that this effect of stress on the co-expression network is robust.

### Importance of Non-Coding Genes in Stress

There was a greater presence of non-coding RNAs after exposure to stress in the RNA-seq network. Non-coding RNAs made up 23% of the set of genes present only in stress, compared to 13% of genes present only in the non-stressed networks. This raises the possibility that the expression of non-coding RNAs becomes more coordinated under stress treatment. An analysis (performed by collaborator Vera Pancaldi) of the non-coding RNAs that appear to be strongly co-regulated only during stress reveals that the majority are annotated antisense RNAs, overlapping protein coding transcripts on the opposite strand. The corresponding protein-coding transcripts (mRNAs) represent a mixture of cell-cycle factors, chromatin remodellers and metabolism related proteins. This suggests these strongly co-regulated non-coding RNAs might play a role in the regula-

tion of these functions during stress. More specifically, edges were classified into three groups: links between two non-coding RNAs, links between two coding RNAs and links connecting one coding transcript to a non-coding one. Figure 5.9 shows the proportion of existing links compared to the total number of possible links within each of these categories, in other words, capturing the density within each of these groups. Stress produces an increase in links connecting the same type of gene (both coding or non-coding) whereas there is no increase in the density of mixed (coding to non-coding) links. This result confirms findings that non-coding antisense RNAs can be regulated independently from their corresponding coding partners [163]. In addition to the antisense RNAs discussed above, some of the non-coding RNAs appearing only in the stressed network are paired with other non-coding RNAs on the opposite strand, while others are intergenic RNAs.

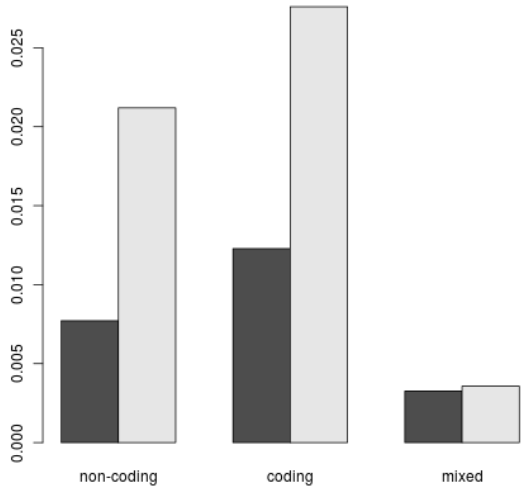


Figure 5.9: The density (existing links over possible links) of coding and non-coding RNA sub-networks in the RNAseq co-expression network. The three categories of links shown are: non-coding to non-coding; coding to coding; and non-coding to coding (mixed). Dark bars shows measures for the non-stressed network, lighter bars shows measures for the stressed network. Stress increases the density of coding to coding and non-coding to non-coding links, without greatly affecting the mixed links. Figure produced by Vera Pancaldi.

### 5.3.2 Protein Interaction Networks

#### Module Overlap

Changes in the co-expression network modularity appear to be translated onto the protein network. In these protein networks, physical interactions between proteins have been assigned a weight according to the estimated probability of the interaction occurring in the stressed or non-stressed condition (see Section

5.2.2 for further details). As shown in Figure 5.10, ModuLand overlap decreases in response to stress for both methods of networks weighting (abundance and co-expression weighting), though this finding is only significant for the abundance weighting (Wilcoxon signed rank test, $p<0.001$ for abundance weighting, $p = 0.6$ for co-expression weighting). For reasons discussed further in Section 5.3.2, the Link Communities algorithm assigns the vast majority of nodes to a single module, making the measure of module overlap largely meaningless.

These effects of stress on the PPI network are less pronounced than in the co-expression networks. Although this result may be a genuine difference between the networks, it could also be due to the relatively small coverage of the PPI network in fission yeast resulting in diminished statistical power to detect stress induced changes.

To test whether a similar change in network structure is also seen in response to other cellular stresses, weighted abundance networks were also constructed from protein abundance data in response to 24 hours of nitrogen starvation (quiescence). As shown in 5.11, the ModuLand overlap is also significantly decreased in response to nitrogen starvation (Wilcoxon signed rank test, $p< 10^{-10}$). Average Link Communities overlap, however, is increased in response to nitrogen starvation (Wilcoxon signed rank test, $p< 10^{-3}$).

As with stress, the Link Communities algorithm assigns the majority of the nodes to a single module, again, complicating the interpretation of the results. It is therefore unclear whether the change in Link Communities overlap represents a difference in the effects of oxidative stress and nitrogen starvation, or is simply due to Link Communities overlap not adequately capturing the overlap in these networks.

It is interesting that, although these two stresses produced different cellular responses, the network effect, as measured by ModuLand overlap, is similar. The potential reasons for the network restructuring – increased robustness, energy saving and development of more distinct functional modules – are plausible responses to both oxidative stress and nitrogen starvation.

#### Shortcoming of the Link Communities Algorithm on PPI networks

Hierarchical clustering in this work was performed using a distance cut-off of 0.4. In PPI networks, hierarchical clustering with a threshold of 0.4 assigned the vast majority of nodes to a single module. At lower cut-off values, all edges were assigned into their own module, essentially meaning that the number of modules a node was assigned to was determined by its degree. A cut-off value for which the clustering did not fall into one of these extremes could not be determined, even when changes to the cut-off were below $10^{-5}$. Further optimization of the cut-off value were prohibited by computation cost.

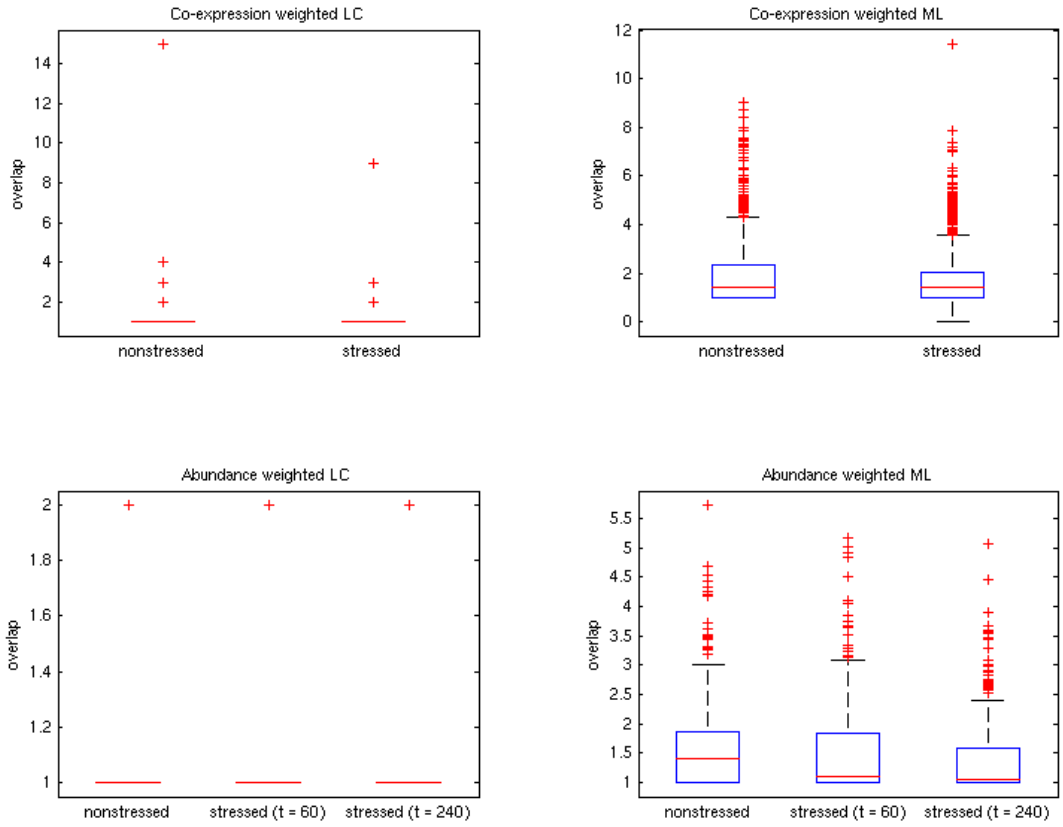


Figure 5.10: Changes to modular overlap in response to oxidative stress (0.5mM hydrogen peroxide). The distinct module finding algorithms were used: ModuLand (ML) and Link Communities (LC, using clustering cut-off of 0.4, see Methods). For ModuLand modules, overlap was measured as ML overlap (see Methods), while for LC modules, overlap was measured as the number of modules a protein belonged to. Average ML overlap decreases from 1.90 to 1.75 for the co-expression weighted networks and from 1.53 to 1.50 (at t= 60min) and 1.33 (at t= 240min) for the abundance weighted networks. Average LC overlap decreases from 1.12 and 1.04 for co-expression weighting and from 1.0043 to 1.0022 (at t= 60 and 240 min) for abundance weighting. Changes in the ML overlap are significant for the abundance weighted network (Wilcoxon signed rank test, $p<0.001$), though not co-expression weighting ($p = 0.5976$).

### Hub-Neighbour Co-Expression

Given the limitations of examining changes in module overlap in the PPI networks, other network measures were used to further quantify the oxidative stress induced changes in network structure.

A 'hub' is a highly connected node in a network - defined, depending on context, either in absolute (for example, more than 5 binding partners [78]) or relative (for example, 5% most connected nodes [17]) terms. It has been suggested that, in PPI networks, hubs are divided into two categories: 'party'

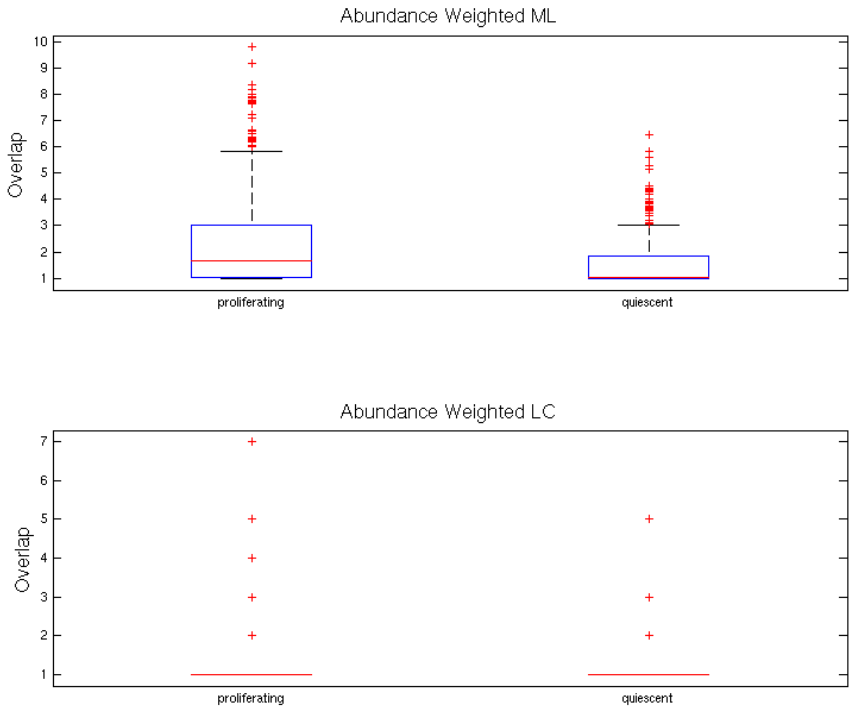


Figure 5.11: Changes to modular overlap in proliferating and quiescent cells. Quiescent cells have been exposed to 24 hours of nitrogen starvation. For ModuLand modules, overlap was measured as ML overlap (see Methods), while for LC modules, overlap was measured as the number of modules a protein belonged to. Average ML overlap decreases from 2.23 to 1.53 while LC overlap increases from 1.08 to 1.13. The decrease in ML overlap is significant, Wilcoxon signed rank test, $p < 10^{-10}$). Note that these boxplots do not capture the size difference in the networks: therefore, though the proliferation network has nodes with higher LC overlap, its average overlap is lower because of a larger number of nodes with LC overlap of 1.

hubs, which are co-expressed with their neighbours, thus binding with most of their partners simultaneously; and 'date' hubs, not necessarily co-expressed with their neighbours and interacting with their partners at different times or locations [17, 78]. Date hubs are thought to function as points of cross-talk between functional modules, while party hubs function within modules.

It is possible that the changes in network modularity could be associated with changes in the date/party behaviour of hubs, which could potentially be observed in the way hubs are co-expressed with their neighbours. The stress induced change in hub-neighbour co-expression was examined for the 2%, 5% and 10% of nodes with highest degree. This corresponded to nodes with degrees greater or equal to 21, 12 and 8 respectively, and 32, 81 and 162 proteins in total. As shown in Figure 5.12, the distribution of hub-neighbour co-expression appears to shift with stress: proteins already highly co-expressed with their neighbours become

increasingly highly co-expressed. This echoes the tighter co-regulation within modules observed in the co-expression network. As a crude way of quantifying this change, a linear regression was fitted to these data. For all three thresholds of hub selection, the slope of the best-fit line was greater than one, indicating that hubs already highly co-expressed with their neighbours become more so after stress. However, this was only significant for the 5% set (for top 2%, 5% and 10% nodes: $R^2$ values were 0.7173, 0.7072 and 0.5889 and 95% confidence intervals on the slope of the best-fit-line were 0.9276 - 1.4263, 1.0171 - 1.2905, and 0.9035 - 1.1074). As the co-expression change is most pronounced at the tail of the distribution (i.e. most highly co-expressed hubs), a linear regression is a very blunt measure of the statistical significance. Unfortunately, the small size of the data set precluded the undertaking of more sophisticated statistical analysis.

In summary, these results hint at an interesting change in hub-neighbour co-expression in response to stress, but are not enough to confidently draw conclusions

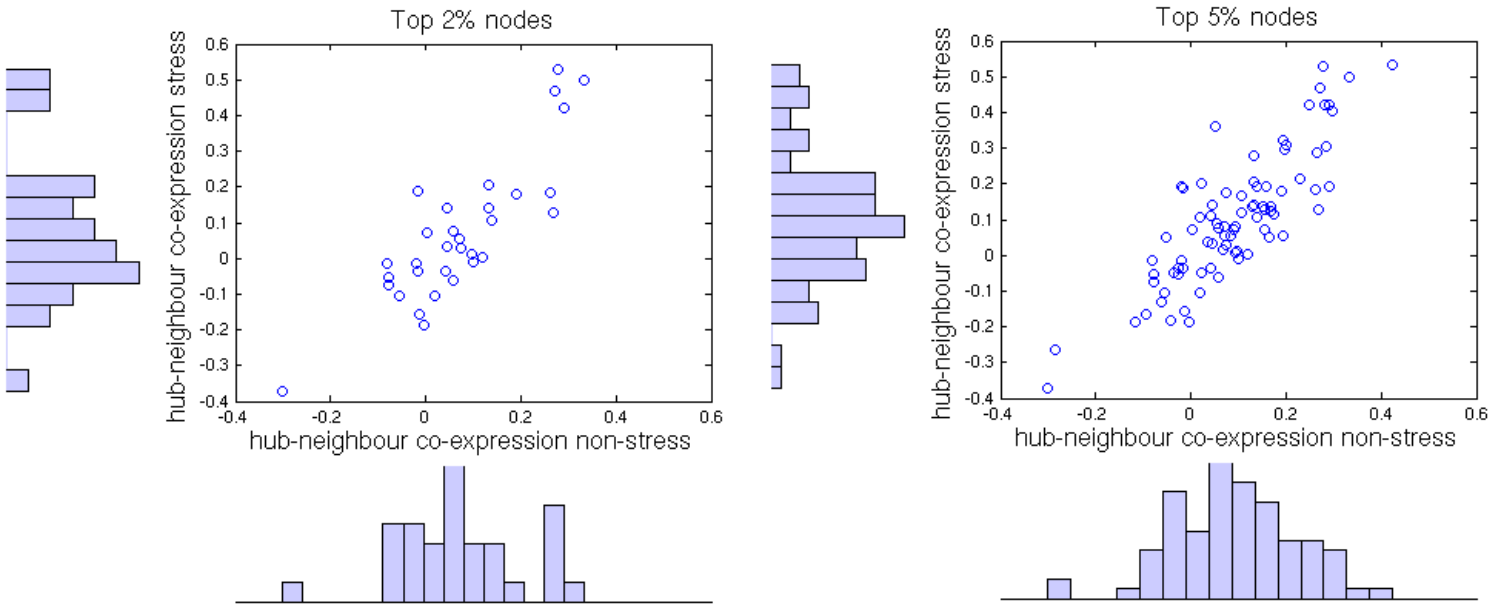


Figure 5.12: The effect of stress on the extent to which hubs are co-expressed with their neighbours. Co-expression values represent the average correlation coefficient (calculated from the RNA-seq data) between a hub (top 2% (left) and top 5% (right) most connected nodes in the iRefIndex PPI network, 32 and 81 proteins, respectively) and its neighbours.

## 5.4 Biological Correlates of Network Change

The results discussed so far have captured global stress induced changes in the structure of co-expression and protein interaction networks. However, it would be interesting to place these changes in the context of biological function, by identifying categories of genes or proteins undergoing the largest shifts in connectivity in response to stress.

### 5.4.1 Principles of Enrichment Analysis

Enrichment analysis is a method of determining whether a specific feature - in this case, GO-category - is significantly over- or under-represented in a gene list, compared to a background gene list. All enrichment analyses presented here were performed using GO::Term-Finder [22], which computes p-values ($p$) using the hypergeometric distribution:

$$p = 1 - \sum_{i=0}^{k-1} \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{i}}$$

where $N$ is the total number of genes in the background list, $M$ is the number of genes with a given annotation in the background list, $n$ is the size of the gene list of interest and $k$ is the number of annotated genes in the gene list of interest.

P-values were corrected using Bonferroni correction for multiple hypothesis testing.

### 5.4.2 Co-Expression Networks

Recall that co-expression network construction involves thresholding edges according to correlation in expression. Consequently, some genes had no connections and were thus not considered part of the network. The presence of a gene in only the stressed or non-stressed network therefore suggests that it is more tightly co-regulated with other genes in one of the conditions.

For the RNA-seq co-expression networks, genes which were present only in the stressed network showed no enrichment for a specific GO-category. The genes present in the unstressed network only were enriched for ion transmembrane transport and related functions (corrected $p < 10^{-4}$) and regulation of nitrogen compound metabolic processes (corrected $p = 0.005$).

No enrichment was found in either set of nodes in the microarray network, which is not surprising. As discussed previously, there is lesser variability between the genetic variants in this data set. This leads to a less accurate estimate of gene co-expression, potentially masking some of the stress induced effects on the network.

Both analyses used the set of genes present in the networks as background to avoid biases towards categories over-represented in the whole network.

To investigate changes in connectivity of genes present in both stressed and unstressed networks, enrichment analysis was performed on the 10% of genes with the greatest stress induced change in degree. In the RNA-seq networks, the set of genes with the greatest stress induced *decrease* in degree was weakly enriched for monosaccharide catabolic processes (corrected p=0.0038). The set of genes with the greatest degree *increase* was enriched for cytoplasmic translation (corrected p=0.00084), suggesting tighter regulation of translation after exposure to stress. Again, the genes present in the network were used as the background set for the analysis. In the microarray co-expression network, no enrichment was found in either set of genes, although when using the whole genome as background, the enrichment for cytoplasmic translation in the genes with increasing degree was recovered (corrected $p< 10^{-17}$).

### 5.4.3 PPI Networks

A similar analysis was performed on PPI networks (only the co-expression weighted, as all genes were present in both networks for abundance weighted networks). Here, the presence of a protein under only one of the conditions is due to all its edges having a weight of zero in the other condition, indicating that the protein is not functionally important in that condition. Neither set of proteins, however, was enriched for any particular GO-category when using the PPI network as background.

Similarly to the co-expression networks, the 10% of proteins undergoing the greatest stress induced change in degree was tested for enrichment. In the PPI networks, unlike the co-expression networks, weighted degree - the sum of the weights of a protein's interaction - was used in this analysis. In these networks, a protein's weighted degree thus represents an approximation of its probability of participating in interactions.

In the co-expression weighted PPI network, the set of proteins with the greatest stress induced *decrease* in weighted degree is enriched (using the rest of the network as background) for mRNA processing and particularly RNA splicing (corrected p<0.0063). In the abundance weighted networks, there is no enrichment using the abundance weighted network as background. However, using either the larger PPI network (that is, not excluding proteins for which no proteomics data was available) or the whole genome as background, the mRNA processing and RNA splicing enrichment is recovered (corrected p<0.0028). The 10% of proteins undergoing the largest degree *increase* were not enriched for any GO-terms in either of the networks using the network as background.

The enrichment analysis in abundance weighted networks was also performed for sets of proteins undergoing a change in centrality in response to stress. Centrality is an alternative method of assessing functional significance in a network, with central genes or proteins generally having greater functional importance [102]. Centrality was measured as betweenness centrality, the number of

shortest path lengths in the network passing through a node. Proteins with decreasing betweenness centrality upon stress treatment are enriched for cytokinesis (corrected $p < 10^{-14}$), while proteins with increasing betweenness centrality are enriched in proteasome subunits (corrected $p < 10^{-19}$). Finally, a group of proteins enriched for cytoskeleton re-organization (corrected $p < 10^{-6}$), showed increased betweenness centrality at the 240 min time-point.

### 5.4.4 Summary of Enrichment Analyses

In summary, these results suggest a stricter control of proteins involved in translation in the stressed condition. Furthermore, stress appears to decrease the involvement of genes related to RNA splicing in interactions. This finding could reflect that rapidly regulated stress-response genes are under-enriched for introns [100], thus leading to a decreased importance of splicing-related proteins during the stress response. This hypothesis is supported by the finding that the enrichment for splicing related categories is no longer present at 4 hours post exposure to stress.

Although the numbers of genes in these lists are small, the enrichment analyses suggest a fundamental role for the proteasome after stress treatment, probably involved in the elimination of the oxidatively damaged protein. Both the enrichment for cytokinesis and cytoskeleton re-organization are likely to be explained by the growth arrest which is initiated during stress response. These findings also suggest an important rearrangement of the cellular structure as a long-term consequence of stress, in line with recent reports of cross-talk between cell cycle and cell shape regulation [233].

## 5.5 Possible Extensions of this Work

The work presented here shows a stress induced restructuring of fission yeast co-expression and weighted protein interaction networks. The results were a lot more pronounced on the co-expression network. While stress induced changes may indeed be more remarkable at the co-expression level, the difference may also be due to the co-expression networks having significantly higher coverage than the PPI networks. A potentially fruitful extension of this work would be to repeat the analyses on networks constructed from predicted PPI data. In addition to increased coverage, this could have the advantage of decreasing systmatic biases in the network.

Additionally, the results presented here suggest changes occur in the way nodes are co-expressed with their neighbours. The mapping of expression dynamics onto protein networks has received considerable attention in the literature [17, 78, 119], making it particularly interesting to investigate the changes seen here further. Furthermore, previous work has suggested that, in response to stress, interactions are pruned to retain only essential ones [152]. In the PPI net-

work generated here, this would translate to nodes going from interacting with multiple partners with relatively uniform probability to having higher probability of interaction with specific partners. A simple approach would be to compare the variance of a node's edge weights before and after stress. A similar approach has been implemented in quantifying the extent of cellular differentiation during development [12].

## 5.6 Conclusion

Gene co-expression networks show higher positive correlation coefficients, longer average shortest path lengths, higher transitivity, and less overlap between modules after exposure to stress. These findings are indicative of a tighter co-regulation between genes within a module, but lesser communication between modules. This type of re-organization might represent the emergence of more specialized functional units in response to stress. It is also consistent with increased network robustness, potentially ensuring resilience to further challenges. Although changes in the weighted PPI networks are more difficult to assess, it appears that the re-organization seen at a gene expression level is indeed translated to the protein level.

Under stress, the co-expression between a group of hubs and their neighbours increases. This change in the hub-partner co-expression distribution is consistent with the strengthening of intra-module connections parallel to a weakening of inter-module links. These findings are reminiscent of a long standing debate about the existence of bimodality in the hub-neighbour co-expression distribution and the distinction between party-hubs (co-expressed with neighbours and binding many partners at once) and date-hubs (not co-expressed with neighbours and binding partners in different places or at different times). However, this data set is not of a sufficient size to justify any claims in this regard.

The analysis presented here also suggests a decreased importance for splicing factors under stress. This effect is observed in two distinct types of protein interaction network: those weighted according to protein abundance as well as those weighted according to protein co-expression. The lesser functional importance of this regulatory mechanism after stress exposure could arise from the need for rapid control of genes in response to stress. Importantly, the phenomenon is no longer seen four hours after exposure to stress, highlighting its association with the transient stage of the transcriptional response. The decreased network centrality of proteins involved in cell division is consistent with the stress-induced growth arrest, while increased centrality of proteasome subunits could indicate a higher turnover of proteins need to eliminate the oxidatively damaged proteins. Finally, increased co-expression between non-coding RNAs in the stressed conditions suggests that they might play an important role in cellular stress response.

# Chapter 6

# Network Resilience to Node Removal: Variability in Network Models and Co-Expression Networks

## 6.1 Introduction

Robustness, the ability to maintain function in the face of perturbation, is considered a key characteristic of evolvable complex systems, including various biological structures [115]. Robustness is interesting as a fundamental biological phenomenon, but also because of its implications for real world applications, such as understanding of disease or the design of new drugs [116]. Networks models have been a popular tool in studying the robustness of various complex systems [27]. It is therefore not surprising that ideas from complex network theory have been used to model the robustness of gene and protein networks.

The use of terminology is not always consistent between the biological and mathematical literature. Thus, before discussing the literature further, it may be useful to clarify some central concepts. In general terms, robustness can refer to tolerance to any type of perturbation. However, in the context of network models, particularly static network models, robustness usually refers to resilience to node removal. The way resilience is measured depends on the network in question: static network models are limited to topological measures of function, typically using global connectivity as a proxy for how well the network is functioning [3]. Dynamic network models, on the other, allow quantifying function more precisely, as the rate of production of certain molecules [30] or the flux through key reactions [113]. These two types of robustness are sometimes referred to as topological and dynamical robustness, respectively [14].

There is considerable literature on various network models' resilience to node removal. The relationship between degree distribution and resilience has been

particularly well documented. As shown by Albert and Barabasi and a number of other authors, random networks, which have Poisson degree distributions, are less tolerant to (random) node removal than networks with the same average degree but a power law degree distribution [3]. On the other hand, networks with power law degree distributions are more vulnerable to targeted removal of high degree nodes. This difference in behaviour is often discussed in terms of a trade-off between average and worst-case behaviour: resilience to random node removal comes at the cost of vulnerability to removal of specific nodes [115].

The relationship between a network's degree distribution and its resilience to node removal is intuitive: in a network with a heavy-tailed degree distribution, removing a random node is likelier to result in the deletion of a node with low degree, which is less likely to have a significant impact on global connectivity. Indeed, for networks with power law degree distributions, it can be shown that maximizing the entropy of the degree distribution (with a constrained average degree) maximizes the network's robustness to node removal [241].

The relevance of these results in terms of biological networks is still debated. Firstly, as discussed in the Introduction Chapter, the previously prevalent idea that power law degree structure is ubiquitous among biological networks is now considered, at least to an extent, the result of systematic biases in the interaction data and flawed statistics [137, 220]. Furthermore, even if heavy-tailed degree distributions are a genuine feature of gene and protein networks, it does not follow that this property is the cause of the robustness of biological systems.

The relationship between network structure beyond degree distribution and robustness to node removal is less clear. There has been some interest in the relationship between a network's modularity (loosely defined as how easily the network is decomposed into separate modules) and its robustness. A highly modular structure is a common feature of biological networks [45, 81, 192, 193, 196]. Some authors have attributed the robustness of biological networks to their high modularity: a highly modularised structure would contribute to robustness by limiting the spread of intra-modular damage to the rest of the network [115].

Contrary to this idea, evidence generally points towards a negative correlation between modularity and robustness to node removal in dynamic network models. Hintze and Adami generated a variety of synthetic metabolic networks through a process of *in silico* evolution which combined random network components and then selected for networks producing key metabolites [84]. Over the course of the evolutionary process, the modularity of the networks, as measured as the presence of bottlenecks in the network, increased. Meanwhile, robustness to both environmental perturbations and node removal decreased, although the decrease was very slight for node removal. Holme constructed synthetic metabolic networks of differing modularity and found that increasing modularity *increased* dynamic robustness to environmental perturbations, but decreased dynamic robustness to node removal [88]. Recently, Tran and Kwon reported

that modularity is negatively correlated with dynamic robustness to node removal in cellular signalling pathways [230]. Thus, taken together, these results suggest a negative association between modularity and dynamic robustness to node removal. The relationship between topological robustness and modularity, on the other hand, remains less well studied.

### 6.1.1 Robustness and Stress

Robustness is a particularly interesting concept in the context of the stress response. The stress response itself maintains the cell's ability to function in the face of external perturbation: it can therefore be considered as an example of robust behaviour. However, as discussed in Chapter 5, the stress response also provides the cell protection against further insults, thus potentially increasing the cell's robustness. Indeed, there has been speculation that the changes observed in cellular networks in response to stress aim to maximize the cell's robustness [152].

Interestingly, it has been suggested there is a fundamental link between robustness and evolvability: not only is robustness selected for during the course of evolution, but a certain degree of robustness is required for a system to be evolvable in the first place [115]. As discussed in Chapter 5, exposure to stress is associated with an increased rate of mutation [57] and greater evolvability [140]. It therefore seems plausible the changes to the co-expression networks in Chapter 5 would be associated with an increase in resilience to node removal.

### 6.1.2 Variability of Resilience

Previous work on network resilience to node removal has focused on comparing worst case behaviour (targeted removal of key nodes) to average or single realisations of random node removal. For example, in Albert and Barabasi's comparison of scale-free and random networks [3], the authors present the average shortest path length of the network after a fraction of the nodes have been removed (Figure 6.1). These results appear to represent the removal a single set of random nodes - giving little information about the *expected* (i.e. average) effect of node removal or the variability of the effect. The same behaviour has also been demonstrated by other others, both analytically and using simulations [27,34,89]. To date however, only a single study has addressed the question of how variable the effect of node removal is [229] and no study has looked at the shape of the distribution in more detail.

### 6.1.3 Aims and Objectives

In this Chapter, we examine the variability of network resilience to node removal in random and scale-free networks. We also examine the effects of stress

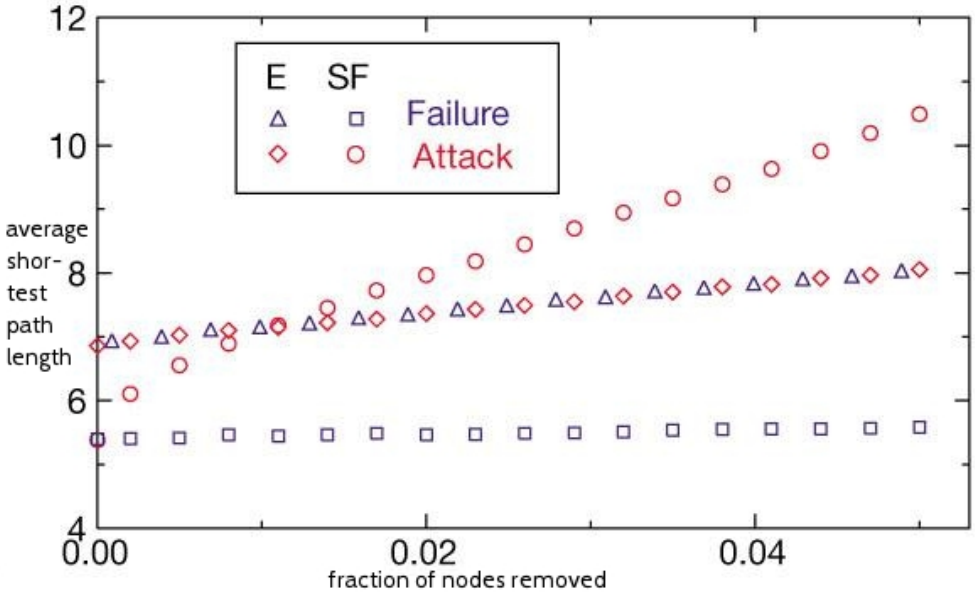


Figure 6.1: The average shortest path length in scale-free (SF) and random (E) networks as a fraction of the nodes are removed in Albert and Barabasi's work [3]. The blue markers ('failure') represent random node removal, while the red markers ('attack') represent targeted removal of high degree nodes. The results suggest scale-free networks are more robust to random node removal, but give no indication of the variability of the effect. Figure adapted from [3].

on resilience to node removal using the co-expression networks constructed in Chapter 5.

## 6.2 Methods

### 6.2.1 Network Models

In line with Albert et al. [3], we compared resilience to node removal in Barabasi-Albert graphs [13] (hence referred to as scale-free (SF) graphs) and Erdős-Rényi (ER) random graphs. Network generation, node removal and path calculations were all implemented using the NetworkX package for Python.

SF networks were generated according to the preferential attachment model: the network is initialized with $m$ nodes and grown one node at the time, until a network with $n$ nodes is reached. Each new node attaches to $m$ of the existing nodes. The probability of attaching to existing node $i$ ($\tau(i)$) is proportional to the degree of $i$ ($\tau(i) = \frac{k_i}{\sum_j k_j}$, where $k_i$ is the degree of $i$).

ER networks were generated by initializing a network with $n$ nodes and then connecting each pair of nodes with probability $p$.

In the work presented here, we used values $n = 1000$ and $m = 2$, giving a SF network with 1000 nodes and 1996 edges. This corresponds to a $p$ of 0.004 ($\frac{1996}{\binom{1000}{2}}$) for ER network generation. Because ER network generation is a probabilistic process, there was slight variation in the number of edges in the

ER network.

### 6.2.2 Stress Networks

We also examined the effect of node removal on the co-expression networks from Chapter 5 before and after exposure to stress. The RNA-seq networks with 40000 edges were used.

### 6.2.3 Resilience Measure

The effect of node removal on the network was measured in terms of network efficiency $e$, given by:

$$e = \frac{1}{\frac{1}{2}n(n-1)} \sum_{i>j} \rho_{i,j}^{-1}$$

with $\rho$ indicating the length of the shortest path between nodes $i$ and $j$. The change in network efficiency in response to node removal was measured by the normalized change in efficiency ($d_f = \frac{e_f - e}{e}$), where $e_f$ is the efficiency after removal of fraction $f$ of the nodes.

Efficiency is a typical choice of measure to quantify the state of the network. Other measures also exist, including average shortest path length and the size of the largest connected component. Average shortest path length becomes problematic for networks with more than one component: the path length between nodes on different components is infinite. Efficiency solves the problem: $\lim_{x\to\infty} \frac{1}{x} = 0$, thus unconnected pairs contribute nothing to the total efficiency. The size of the largest connected component is typically used in theoretical work (percolation models, for example [27]), particularly in the context of finding the 'percolation threshold', the fraction of nodes that can be removed before the network fragments into multiple disconnected components. In the context of gene and protein networks, this measure has the disadvantage of not giving information about the connectivity within the largest component. Efficiency captures this information, but should also be able to detect network fragmentation: studies looking at both efficiency and the size of the largest connected component have found these measures give similar results [229].

## 6.3 Network Models

Figure 6.2 compares the change in efficiency after node removal in a SF and ER network, up to removal of 10% of the nodes (corresponding to removal of 100 nodes), for 1000 realisations of random node removal. As expected, the mean loss of efficiency is greater in the ER network ($< d_{f=0.1} >= -0.0447$ for the SF network and $< d_{f=0.1} >= -0.0581$ for the ER network). The variability of the response is greater in the SF network: the standard deviation of $d_{f=0.1}$ in the SF network was 0.0231, and 0.0098 in the ER network. These results are in

line with those of Trajanovski et al [229]. We also examine the 'skewness', or the symmetry of the distribution, defined as $s = \frac{E(x-\mu)^3}{\sigma^3}$ where $\mu$ is the mean of $x$, $\sigma$ its standard deviation and $E(t)$ is the expected value of $t$. A negative skewness indicates a long tail at low values, a positive skewness indicates a long tail at high values. The skewness of the distribution for the SF and ER network is -1.08 and -0.071 respectively, indicating the SF network has a longer tail at at low $d_{f=0.1}$ values.

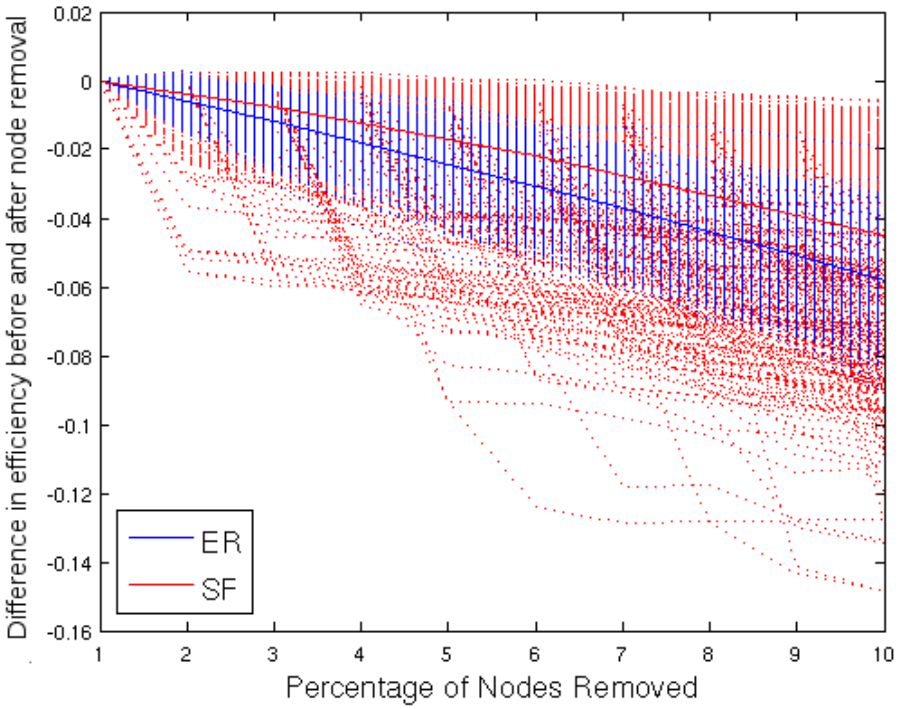


Figure 6.2: Change in efficiency in response to removal of an increasing proportion of the nodes in a SF and ER network. Each dotted line represents one realisation of random node removal, with a total of 1000 realisation for each network. The solid lines represent average behaviour across realisations.

Next, we examined the behaviour of 100 networks of both types in order to confirm the behaviour we observe is a general property of networks of this type. Figure 6.3 shows the distribution of the change in efficiency after removal of 10% of the nodes for 500 realisations of random node removal for each network. This confirms the previous results: SF networks have greater average resilience, but the response is more variable, with a long left tail.

## 6.4 Stress Networks

One of the possible explanations put forward for the stress induced changes in network structure seen in Chapter 5 is that the post-stress network is more resistant to further damage [152]. We sought to explicitly test this by examining how node removal affects network efficiency before and after exposure to stress in the

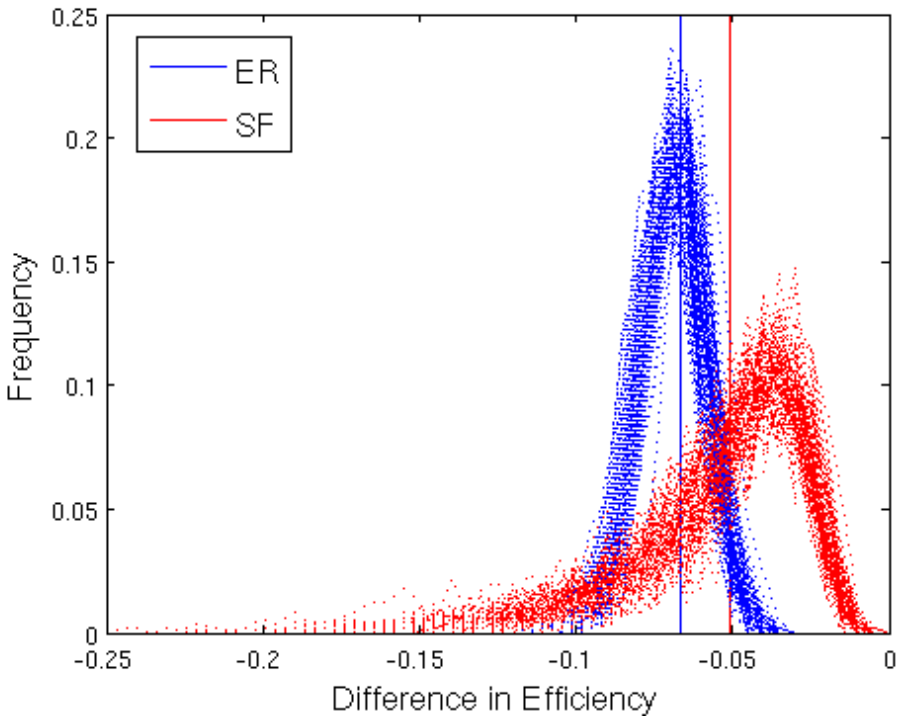


Figure 6.3: Distribution of the change in efficiency after removal of 10% of the nodes for 500 realisations of random node removal for SF and ER networks. Each line corresponds to the distribution of $d_{f=0.1}$ scores for a single network. The horizontal lines indicate average change in efficiency across the 100 networks and 500 realizations.

RNA-seq co-expression networks. The RNA-seq networks were chosen instead of the microarray networks because of the methodological problems associated with the microarray networks.

First, we investigate network response to removal of a single node. For both of the RNA-seq networks, we computed the normalized difference between efficiency before and after single node removal. While the mean $d_{n=1}$ (using $n$ to indicate how many nodes have been removed) score was not different for the two networks ($-1.52 \times 10^{-4}$ and $-1.62 \times 10^{-4}$ for pre and post stress respectively, p = 0.11, ranked sum test), the standard deviation in the post-stress network was greater ($5.06 \times 10^{-4}$ vs $7.18 \times 10^{-4}$) and the skewness more negative (-2.81 vs -5.06). This suggests that while the average robustness of the network remains the same after exposure to stress, the variability of the effect of removing a single node is greater, with a greater probability of exceptionally high damage.

To examine whether this behaviour was simply due to the change in degree distribution or the change in the number of nodes, we computed 10 networks with the same degree structure as the original pre and post stress networks by reshuffling the networks' edges. In these networks, the effect of stress on average $d_{n=1}$ is slightly greater (-1.45×$10^{-4}$ vs -1.95×$10^{-4}$ pre and post stress respectively), but the difference in standard deviation is smaller (4.94×$10^{-4}$ vs

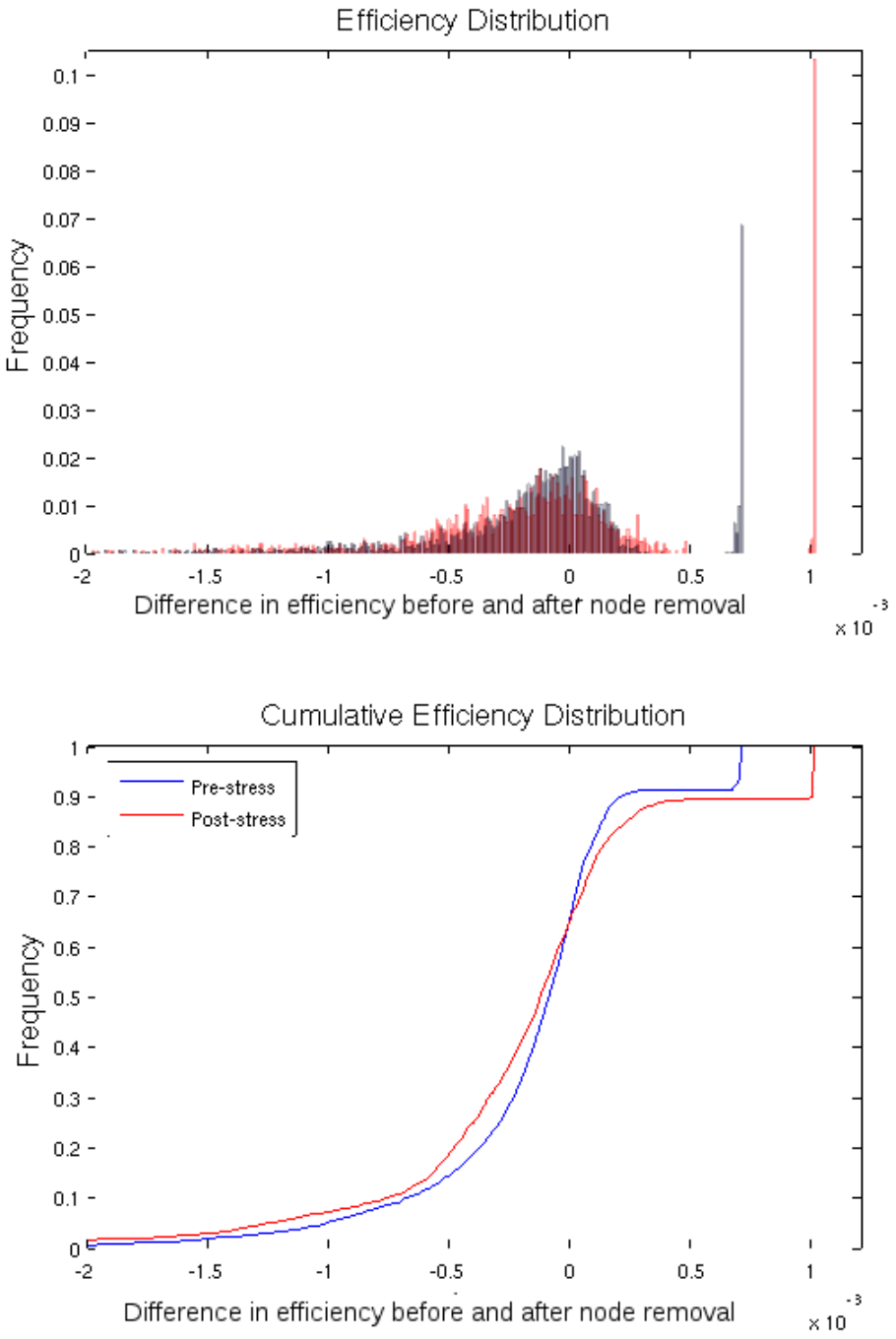


Figure 6.4: Robustness to random node removal in RNAseq co-expression networks, as measured by change in efficiency. The figures shows the distribution (top) and cumulative distribution (bottom) in normalized difference in efficiency before and after node removal, for each node in the network.

$6.08 \times 10^{-4}$) and the skewness becomes less, not more, negative (-3.76 vs -2.85). Thus, changes in degree distribution do not fully explain the effect of stress on the network's resilience to node removal. This result is compatible with the idea

the changes in modularity structure in response to stress increase the network's robustness - although this behaviour may also relate to other structural changes beyond the degree distribution.

Next, we adopted the approach used in the analysis of the SF and ER network models: random nodes were sequentially removed from the network, up to deletion of 10% of the network's nodes (corresponding to 284 nodes in the pre-stress and 198 nodes in the post-stress network). In one sense, this is a more biologically realistic model of damage to proteins, as environmental perturbations would be unlikely to selectively cause loss of function in all copies of a single protein. The model remains somewhat unrealistic, as 10% of the nodes are completely removed (corresponding to losing all existing copies of the protein), with no damage to other proteins. However, this is a necessary approximation, as our networks cannot represent node damage, only full removal of a node.

Figure 6.5 shows the resilience to node removal in the co-expression network before and after exposure to stress, for 200 realisations of sequential random node removal. The average normalised change in efficiency was -0.045 in the pre-stress network and -0.035 in the post-stress network, while the standard deviation was 0.0082 and 0.0218 and the skewness -0.2174 and -6.9442 in the pre and post-stress networks respectively. Thus, while on average the post-stress network is more resilient to node removal, the variability of the damage is greater and extreme damage is more likely.

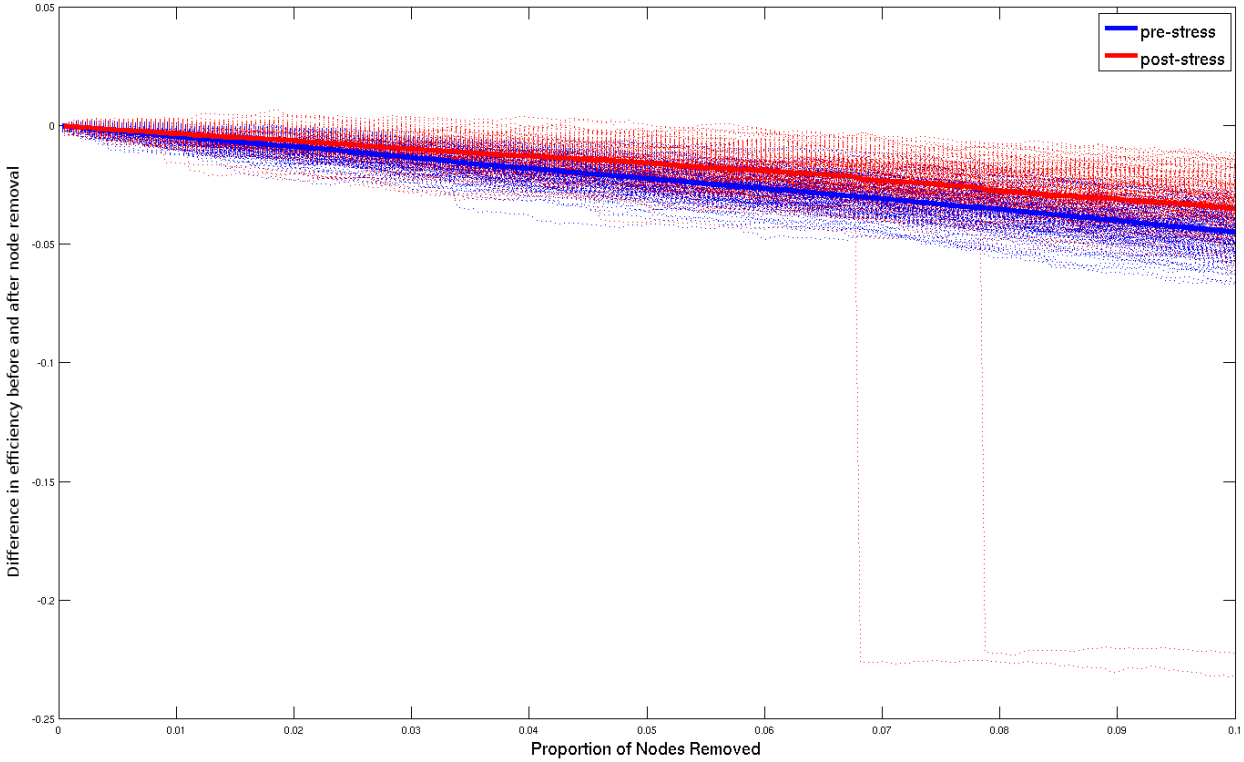


Figure 6.5: Change in network efficiency in response to random node removal in co-expression networks before (blue lines) and after (red lines) exposure to stress. The dotted lines represent a single realisation of random node removal (total of 200 realisations) and solid lines represent average behaviour. On average, the post-stress network experiences lesser damage. The variability of the response is greater in the post-stress network, with occasional extreme changes in efficiency.

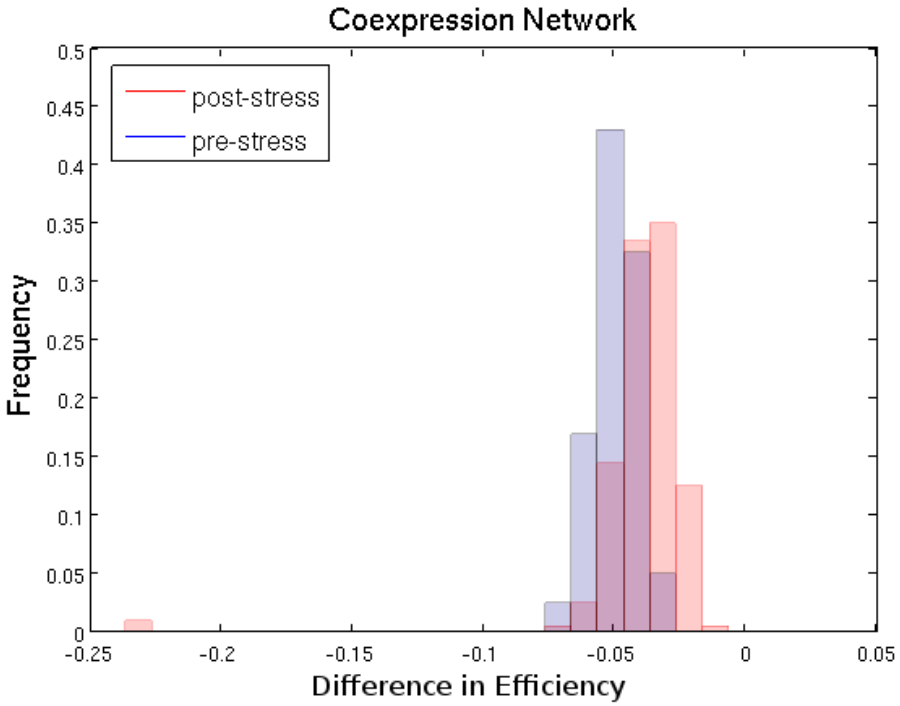


Figure 6.6: Distribution of change in network efficiency after removal of 10% of nodes in the network (corresponding to 284 nodes in the pre-stress and 198 nodes in the post-stress network).

A drawback of this model is that only nodes included in the network can be deleted. This is unrealistic: random mutations would not only target highly co-expressed genes and proteins. To correct for this, both the single and 10% node removal simulations were re-run using a network covering the whole genome. Genes not part of the original networks (i.e not highly co-expressed with any other gene) were considered as nodes with no connections.

For both simulations, the pattern of behaviour remains the same: the post-stress network is, on average, less damaged by removal of a node, but the standard deviation of the distribution is greater and the skewness is more negative. Specifically, when looking at the effect of removing a single node, the mean $d_{n=1}$ pre- and post was -7.34×$10^{-5}$ vs -5.43×$10^{-5}$ respectively, the standard deviation was 5.54×$10^{-4}$ vs 6.93×$10^{-4}$ and the skewness was -45 vs -169. When removing 10% of the nodes, the mean $d_{f=0.1}$, pre- and post was -0.046 vs -0.036 respectively, the standard deviation was 0.013 vs 0.034 and the skewness was -0.13 vs -4.56. This difference in behaviour pre and post exposure to stress is illustrated in Figure 6.7.

Again, these results show that while the expected effect of node removal on the post-stress network is smaller, there is a greater incidence of extreme loss of network function after exposure to stress.

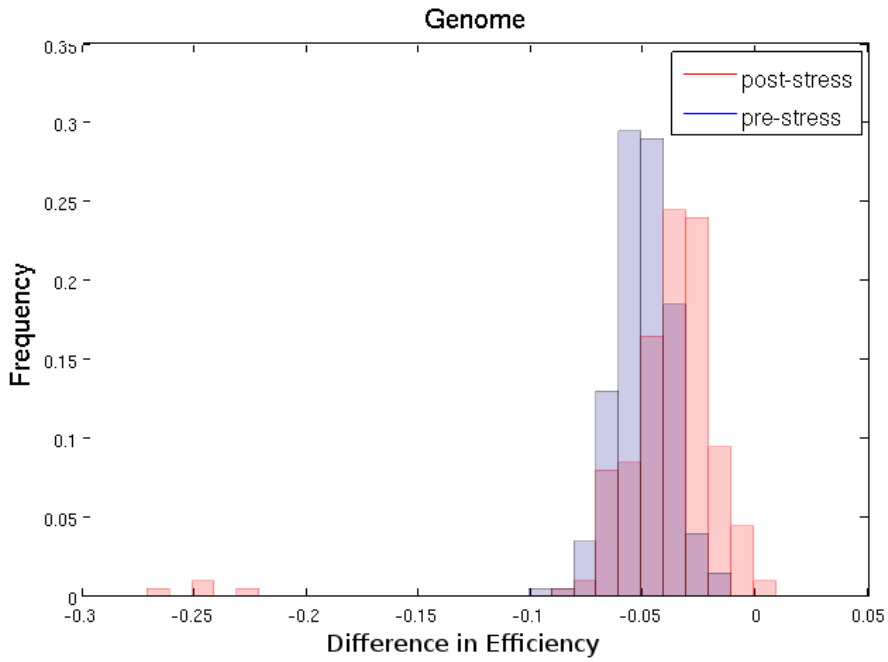


Figure 6.7: Distribution of change in network efficiency after removal of 10% of genes from the whole genome networks (588 genes out of the full 5883 gene genome).

## 6.5 Discussion and Conclusion

Our results indicate that while the overall expected (average) resilience to node removal is greater in scale-free networks than in random networks, the *variability* of the response is greater in scale-free networks. This result is not unexpected: it is an extension of the result that scale-free networks are more vulnerable to the targeted removal of high degree nodes than random networks. However, the shape of the response distribution had not previously been explored (although, while this work was being undertaken, work relating to this idea was published by Trajanovski and colleagues [229]).

We also examine the variability of the effect of random node removal in the co-expression networks from Chapter 5. Exposure to stress causes the co-expression network to behave more like the scale-free network: both the expected resilience to node removal and the variability of the response are greater after exposure to stress. Furthermore, the incidence of extreme loss of efficiency is higher after exposure to stress. To some extent, this change may be attributable to the change in degree distribution or network size. However, similar changes are not seen when simply producing networks with the same degree distribution as the pre and post networks - thus, degree distribution alone is not enough to explain the behaviour.

The results presented here suggest that, after exposure to stress, yeast cells will be, on average, more resistant to deleterious mutations or protein damage. However, the variability of the effects of deleterious mutations or protein damage

will also be greater, with a greater incidence of extremely deleterious events. This hypothesis is potentially experimentally testable.

From a population genetics point of view, a potential explanation for the behaviour we observe is the relationship between stress and evolution: exposure to cellular stress increases mutation rate [57]. Increased resilience to mutation would therefore allow cells to tolerate the higher frequency of deleterious mutations but also enable stressed cells to explore a greater range of new, potentially adaptive, phenotypes.

It is also possible to interpret the changes we see on a single cell level: they may represent a risk management strategy. If a cell exists in an environment where even moderate malfunction is likely to significantly impair survival, it is better to opt for minimal damage most of the time, at the cost of occasional catastrophic failure. On the other hand, if the cell exists in a safer environment, where even high levels of damage do not significantly affect the probability of survival, the better strategy is to avoid very high levels of damage, even at the cost of higher average damage. These two modes of behaviour correspond to those seen in the post-stress and pre-stress co-expression networks respectively.

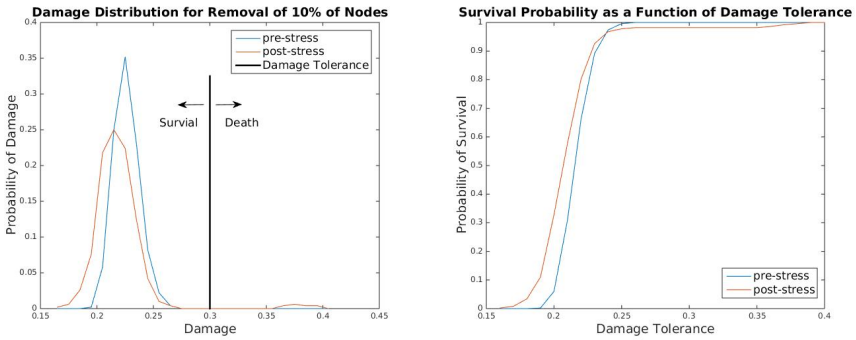


Figure 6.8: Illustration of how different network structures may be beneficial in different environments. The left hand panel shows the damage distribution (i.e. change in network efficiency) $p(d)$ for 10% node removal for the pre- and post-stress networks. The damage tolerance $t$, capturing the level of network damage the cell can sustain before dying is also illustrated. The right hand panel shows the relationship between overall survival probability $r = \int_{d=0}^{t} p(d)$ and the damage tolerance for both networks. For high damage tolerance (i.e. unchallenging environments where the cell is able to survive despite a high level of network damage), the pre-stress network has a higher overall survival probability than the post-stress network because of the occasional catastrophic failures of the post-stress network. However, if the damage tolerance is low, the pre-stress network performs better than the post-stress network. Thus, the shape of the damage distribution can have important consequences for the cell.

In more general terms, assuming the cell is attempting to maximize its chances of survival, we can equate the probability of survival to the concept of *utility* from economics. Thus, the relationship between damage and survival probability can be thought of as a utility function. The shape of this utility function determines the optimal strategy to adopt: concave utility functions

promote risk averse strategies (low variance) while convex utility functions lead to risk-seeking behaviour (high variance). Similarly, as illustrated in Figure 6.8, we can hypothesise the change in the variance and skewness of the $d_f$ distribution relates to the change in the relationship between $d_f$ and probability of survival.

# Chapter 7

# Discussion

This thesis has examined the use of network approaches in drawing meaning from a rapidly increasing volume of biological data. We have explored network models in the context of three biological applications: the prediction of protein function, the study of loss of function variation in the human genome and the representation of the effects of stress in fission yeast cells. This chapter will briefly summarise the conclusions of each of the chapters, suggest directions for further work and briefly discuss some overarching themes.

## 7.1 Protein Function Prediction

Chapter 2 addresses the use of guilt-by-association approaches in predicting protein function. We develop a novel prediction algorithm (Compass), based on graph kernel and dimensionality reduction approaches and compare it to a leading network based prediction algorithm (GeneMANIA) on a number of benchmarks. The relative performance of the two methods depends on the benchmark, although Compass outperforms GeneMANIA on a majority of the cases.

We also explicitly examine potential biases in GO-based comparisons of prediction algorithms (such as the CAFA challenge). We create a 'roll-back benchmark' in which we make predictions based on data available at a specific date and use new annotations made after this date to evaluate performance, thus mimicking the CAFA challenge. Unsurprisingly, we find that both GeneMANIA and Compass predict annotations for high degree nodes more successfully than for low degree nodes. We also find that in the yeast (but not fly) benchmark, annotations acquired shortly after the cut-off date 1) correspond to genes with higher degree and 2) are easier to predict than later annotations. We hypothesise that high performance on these 'early' annotations is (at least partially) related to their high degree.

These effects could reflect a systematic bias in how annotations are acquired: it is reasonable to suggest that genes with well characterized interactions are more likely to become functionally well characterized in the near future. This bias would translate into the benchmark: because the time window between

prediction and evaluation in CAFA is relatively short (6 months), algorithms favouring high degree genes will appear to perform well, because they mimic the process of label acquisition. Our roll-back benchmark only covers two organisms: yeast and fly - it is therefore unclear whether these effects are simply a property of the yeast dataset or whether they reflect a more general trend. Extending the roll-back benchmark to include further organisms could potentially answer this question.

CAFA-style benchmarks are appealing because of their efficiency: instead of performing specific experiments to test the predictions, the process relies on the continuous acquisition of novel annotations. However, this system only provides a fair assessment of prediction methods if the process of label acquisition is independent of the prediction algorithms. As discussed above, our results suggest this may not be the case: both network-based prediction and label acquisition depend on a gene's degree. This concern, however, is not limited to the benchmarking of network-based algorithms: similar problems arise for sequence or structural similarity-based methods, because sequence and structural similarity drive label acquisition. Overall, these effects would lead CAFA-style benchmarks to favour the methods mimicking the label acquisition over methods providing other forms of insight.

One solution to this problem is to perform specific experiments to test predictions instead of relying on the process of label acquisition to provide true positives. There is a concern, however, that this type of scheme could potentially be too expensive for a large-scale evaluation of multiple methods making predictions for multiple functional categories. Another solution would be to develop more benchmarks similar to our phenotypic benchmark, where the true positives are derived from genome-wide screens, thus circumventing the biases associated with label acquisition. Publicly available repositories of this type of data (such as the Genome RNAi database used in our work) would greatly facilitate the implementation of these types of benchmark.

When thinking about both algorithm and benchmark design, the appropriate scope for our prediction methods emerges as a key question. It is tempting to seek a 'one size fits all' algorithm as a general tool for protein function prediction. However, the results in this thesis suggest that optimal algorithm choice may depend on the biological context of the prediction. For example, an algorithm which performs well for large and diverse functional groups may not be the best choice for prediction in a very narrow functional context. For those interested in specific biological problems, it may therefore prove more effective to tailor the design and benchmarking of prediction algorithms to the problem at hand.

It is important to note, however, that this function-specific tailoring approach is not appropriate for all prediction problems. One of the key uses of function prediction is the automated annotation of genes currently lacking any functional labels. Clearly, for this type of prediction problem, we are unable to tailor our

algorithms (or our benchmarking) to a specific functional context.

Nevertheless, there are other ways in which prediction methods can be optimized for unannotated genes. For example, this type of prediction problem has been challenging for network-based methods because there is very often little functional association data for unannotated genes. While this thesis only considered kernels derived from functional association networks, kernel approaches are not limited to this data: it is straightforward to generate kernels based on other types of similarity. It may therefore prove fruitful to exploit the kernel-based algorithms discussed in this thesis using more diverse data sources.

## 7.2 Loss of Function Variation

In Chapter 3, we seek to identify genetic interactions between loss of function tolerant genes, based on how often non-functional variants of these genes appear in healthy genomes. We first predict interacting gene pairs using a hyper-geometric model and then develop a modularity based approach to identify groups of potentially interacting genes. None of the putative interactions we identify correspond to known interactions, although a few are promising candidates. This does not necessarily indicate our approach is flawed: genetic interaction data in human is sparse and the number of genomes available for our analysis was small. The number of sequenced genomes is growing rapidly - the methods developed in this chapter will be applied to a larger dataset once this becomes available.

There are a number of reasons why identifying genetic interactions in the human genome is interesting. Firstly, a human genetic interaction network would provide a valuable additional resource for the types of network-based analyses (protein function prediction for example) discussed in this thesis. Perhaps more importantly, however, identification of genetic interactions could help solve a key puzzle in modern biology: the *missing heritability problem.*

Understanding the genetic factors which control susceptibility to disease is important because this knowledge can inform the diagnosis, prevention and treatment of disease. Genome-wide association studies (GWAS) have allowed identification of variants associated with disease phenotypes. Interestingly, however, the variants identified so far fail to fully explain the familial clustering of the phenotypes: based on the disease variants identified in GWAS, we would expect the studied diseases to be less heritable than they appear to be.

A number of explanations have been suggested for this missing heritability, including incomplete characterisation of disease variants and shared environmental factors contributing to the apparent heritability of diseases. Recently, the presence of genetic interactions has been advanced as an alternative explanation. Statistical models of heritability assume no interaction between genes - the presence of disease associated genetic interactions could therefore be the key to the unexplained heritability. Identifying such disease associated interactions

through GWAS requires very large sample sizes. The methods presented in this thesis could therefore provide a valuable alternative approach.

In Chapter 4, we build a network based three-class classifier to differentiate between LoF tolerant genes and recessive and dominant disease genes, combining network centrality and guilt-by-association approaches. We find that best results are achieved by combining PPI degree data with STRING functional association information. Unlike some previous studies, we find that centrality in metabolic networks is not significantly different in the three gene classes. We also find that the inclusion of genetic interaction data does not improve the performance of our predictor. We only consider a nearest neighbour classifier - it is possible that more sophisticated prediction approaches might further improve performance or allow inclusion of the genetic interaction data. The main conclusion of this chapter is that guilt-by-association can be used to predict functional importance. Collaborators will extend this work by integrating guilt-by-association approaches into an existing prediction pipeline.

The work in Chapter 4 focuses on the action of individual genes. As outlined above, however, interactions between different genes may play an important role in disease susceptibility. Functional association networks naturally lend themselves to the study of interaction effects. It would therefore be valuable to extend these types of network-based predictors beyond the effects of single genes.

## 7.3 Stress Response

Chapter 5 examines changes in fission yeast co-expression and PPI networks after exposure to oxidative stress. We find co-expression networks re-organize towards a more modularised structure: while sets of genes become more tightly co-expressed, co-expression between these modules is decreased. A similar change is also found in the structure of weighted protein interaction networks in response to both oxidative stress and nitrogen starvation, confirming and extending previous findings. These changes in network structure could represent the emergence of more specialised functional modules, an increase in network robustness and/or result from energy saving measures. Additionally, stress is found to induce tighter co-regulation of non-coding RNAs, decreased functional importance of splicing factors, as well as changes in the centrality of genes involved in cytoskeleton organization, cell division, and protein turnover.

In Chapter 6, we address the idea that the stress-induced changes in the co-expression network might correspond to an increase in cellular robustness. Using decrease in network efficiency as an indicator of loss of network functionality, we study the effect of node removal, before and after exposure to stress. We find that after exposure to stress, the average decrease in efficiency is smaller, but the variance of the response is greater. The increased robustness to node removal

may reflect increased tolerance of loss of function mutations, which would be in line with increased evolvability of cells after exposure to stress.

There is increasing interest in comparing the topology of gene and protein networks under different conditions. As the availability of condition-specific data increases, comparative network analysis will become increasingly important. In some ways, this type of analysis is more straightforward than attempting to interpret the properties of a single network. Whether network properties such as degree distribution or clustering coefficient appear 'surprising' and thus potentially meaningful is crucially dependent on the choice of null model. As illustrated by the example of clustering coefficients in coexpression networks, inappropriate choice of null can lead to attributing meaning to trivial network properties. Comparative network analysis avoids this issue: the comparison is between the two conditions, thus avoiding the need to choose a null model. On the other hand, however, comparative network analysis faces other challenges. For example, statistical methods for untangling random effects from genuine condition-related changes in network structure are not well established. There is clearly a need for further development of statistical tools in the context of comparative network analysis.

## 7.4 Overall Conclusions

While the work presented in this thesis corresponds to three specific biological scenarios, it is interesting to attempt to identify some overarching ideas. Part of the appeal of network models is that they provide a unifying framework for working with data from a multitude of heterogeneous sources. However, several of our findings highlight the importance of considering both how the network is generated and the biological context it is used in. We have already discussed this idea in relation to the work in Chapter 2. The development of the network clustering approaches in Chapter 3 also illustrates this principle: a general network algorithm was outperformed by one explicitly modelling the network generation process. As 'big data' and machine learning are becoming increasingly central in biological research, the role of networks as a tool for data integration is growing in importance. It will be interesting to see whether it is possible to develop network based integrative models that also exploit our understanding of the properties of particular data sources.

# Bibliography


[1] 1000 Genomes Project Consortium et al. An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56–65, 2012.

[2] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, Aug. 2010.

[3] R. Albert, H. Jeong, and A.-L. Barabasi. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, July 2000.

[4] C. Alfarano, C. E. Andrade, K. Anthony, N. Bahroos, M. Bajec, K. Bantoft, D. Betel, B. Bobechko, K. Boutilier, E. Burgess, K. Buzadzija, R. Cavero, C. D'Abreo, I. Donaldson, D. Dorairajoo, M. J. Dumontier, M. R. Dumontier, V. Earles, R. Farrall, H. Feldman, E. Garderman, Y. Gong, R. Gonzaga, V. Grytsan, E. Gryz, V. Gu, E. Haldorsen, A. Halupa, R. Haw, A. Hrvojic, L. Hurrell, R. Isserlin, F. Jack, F. Juma, A. Khan, T. Kon, S. Konopinsky, V. Le, E. Lee, S. Ling, M. Magidin, J. Moniakis, J. Montojo, S. Moore, B. Muskat, I. Ng, J. P. Paraiso, B. Parker, G. Pintilie, R. Pirone, J. J. Salama, S. Sgro, T. Shan, Y. Shu, J. Siew, D. Skinner, K. Snyder, R. Stasiuk, D. Strumpf, B. Tuekam, S. Tao, Z. Wang, M. White, R. Willis, C. Wolting, S. Wong, A. Wrong, C. Xin, R. Yao, B. Yates, S. Zhang, K. Zheng, T. Pawson, B. F. Ouellette, and C. W. Hogue. The biomolecular interaction network database and related tools 2005 update. *Nucleic Acids Research*, 33(Database issue):D418–424, Jan. 2005.

[5] J. D. Allen, Y. Xie, M. Chen, L. Girard, and G. Xiao. Comparing statistical methods for constructing large scale gene networks. *PLoS One*, 7(1):e29348+, Jan. 2012.

[6] P. Aloy and R. B. Russell. Interrogating protein interaction networks through structural biology. *Proceedings of the National Academy of Sciences*, 99(9):5896–5901, Apr. 2002.

[7] P. Aloy and R. B. Russell. InterPreTS: protein interaction prediction through tertiary structure. *Bioinformatics*, 19(1):161–162, Jan. 2003.

[8] D. Altshuler, M. J. Daly, and E. S. Lander. Genetic mapping in human disease. *Science*, 322(5903):881–888, Nov. 2008.

[9] R. Anglani, T. M. Creanza, V. C. Liuzzi, A. Piepoli, A. Panza, A. Andriulli, and N. Ancona. Loss of connectivity in cancer Co-Expression networks. *PLoS One*, 9(1):e87075+, Jan. 2014.

[10] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris,

D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nature Genetics*, 25(1):25–29, May 2000.

[11] M. Babu, J. Vlasblom, S. Pu, X. Guo, C. Graham, B. D. Bean, H. E. Burston, F. J. Vizeacoumar, J. Snider, S. Phanse, et al. Interaction landscape of membrane-protein complexes in saccharomyces cerevisiae. *Nature*, 489(7417):585–589, 2012.

[12] C. R. Banerji, D. Miranda-Saavedra, S. Severini, M. Widschwendter, T. Enver, J. X. Zhou, and A. E. Teschendorff. Cellular network entropy as the energy potential in waddington's differentiation landscape. *Scientific Reports*, 3, Oct. 2013.

[13] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, Oct. 1999.

[14] A.-L. Barabasi and Z. N. Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, Feb. 2004.

[15] M. Barrios-Rodiles, K. R. Brown, B. Ozdamar, R. Bose, Z. Liu, R. S. Donovan, F. Shinjo, Y. Liu, J. Dembowy, I. W. Taylor, V. Luga, N. Przulj, M. Robinson, H. Suzuki, Y. Hayashizaki, I. Jurisica, and J. L. Wrana. High-Throughput mapping of a dynamic signaling network in mammalian cells. *Science*, 307(5715):1621–1625, Mar. 2005.

[16] K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano. Reverse engineering of regulatory networks in human b cells. *Nature Genetics*, 37(4):382–390, Apr. 2005.

[17] N. N. Batada, T. Reguly, A. Breitkreutz, L. Boucher, B.-J. Breitkreutz, L. D. Hurst, and M. Tyers. Stratus not altocumulus: A new view of the yeast protein interaction network. *PLoS Biology*, 4(10):e317+, Sept. 2006.

[18] E. A. Bender and Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3):296–307, May 1978.

[19] D. B. Berry and A. P. Gasch. Stress-activated genomic expression changes serve a preparative role for impending stress in yeast. *Molecular Biology of the Cell*, 19(11):4580–4587, Nov. 2008.

[20] C. Blaschke, R. Hoffmann, J. C. Oliveros, and A. Valencia. Extracting information automatically from biological literature. *Comparative and Functional Genomics*, 2(5):310–313, Oct. 2001.

[21] L. Bonetta. Protein-protein interactions: Interactome under construction. *Nature*, 468(7325):851–854, Dec. 2010.

[22] E. I. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J. M. Cherry, and G. Sherlock. GO::TermFinder–open source software for accessing gene ontology information and finding significantly enriched gene ontology terms associated with a list of genes. *Bioinformatics*, 20(18):3710–3715, Dec. 2004.

[23] K. R. Brown and I. Jurisica. Online predicted human interaction database. *Bioinformatics*, 21(9):2076–2082, May 2005.

[24] A. Brückner, C. Polge, N. Lentze, D. Auerbach, and U. Schlattner. Yeast Two-Hybrid, a powerful tool for systems biology. *International Journal of Molecular Sciences*, 10(6):2763–2788, June 2009.

[25] M. Buljan, G. Chalancon, S. Eustermann, G. P. Wagner, M. Fuxreiter, A. Bateman, and M. M. Babu. Tissue-Specific splicing of disordered segments that embed binding motifs rewires protein interaction networks. *Molecular Cell*, 46(6):871–883, June 2012.

[26] G. Caldarelli, A. Capocci, D. L. Rios, and M. A. M. Noz. Scale-Free networks from varying vertex intrinsic fitness. *Physical Review Letters*, 89(25):258702+, Dec. 2002.

[27] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters*, 85(25):5468–5471, Oct. 2000.

[28] P. Chebotarev. Spanning forests and the golden ratio. *Discrete Applied Mathematics*, 156(5):813–821, Mar. 2008.

[29] B.-R. Chen, Y. Li, J. R. Eisenstatt, K. W. Runge, and S. Whitehall. Identification of a lifespan extending mutation in the schizosaccharomyces pombe cyclin gene clg1+ by direct selection of long-lived mutants. *PloS One*, 8(7):e69084, 2013.

[30] W. W. Chen, B. Schoeberl, P. J. Jasper, M. Niepel, U. B. Nielsen, D. A. Lauffenburger, and P. K. Sorger. Input–output behavior of erbb signaling pathways as revealed by a mass action model trained against dynamic data. *Molecular Systems Biology*, 5(1):239, 2009.

[31] X. Chen, M. Chen, and K. Ning. BNArray: an r package for constructing gene regulatory networks from microarray data by using bayesian network. *Bioinformatics*, 22(23):2952–2954, Dec. 2006.

[32] H. N. Chua, W.-K. Sung, and L. Wong. Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions. *Bioinformatics*, 22(13):1623–1630, July 2006.

[33] H.-Y. Y. Chuang, E. Lee, Y.-T. T. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Molecular Systems Biology*, 3(1), Oct. 2007.

[34] R. Cohen, K. Erez, D. ben Avraham, and S. Havlin. Resilience of the internet to random breakdowns. *Physical Review Letters*, 85(21):4626–4628, Nov. 2000.

[35] H. J. Cordell. Epistasis: what it means, what it doesn't mean, and statistical methods to detect it in humans. *Human Molecular Genetics*, 11(20):2463–2468, Oct. 2002.

[36] M. Costanzo, A. Baryshnikova, J. Bellay, Y. Kim, E. D. Spear, C. S. Sevier, H. Ding, J. L. Y. Koh, K. Toufighi, S. Mostafavi, J. Prinz, R. P.

St. Onge, B. VanderSluis, T. Makhnevych, F. J. Vizeacoumar, S. Alizadeh, S. Bahr, R. L. Brost, Y. Chen, M. Cokol, R. Deshpande, Z. Li, Z.-Y. Lin, W. Liang, M. Marback, J. Paw, B.-J. San Luis, E. Shuteriqi, A. H. Y. Tong, N. van Dyk, I. M. Wallace, J. A. Whitney, M. T. Weirauch, G. Zhong, H. Zhu, W. A. Houry, M. Brudno, S. Ragibizadeh, B. Papp, C. Pál, F. P. Roth, G. Giaever, C. Nislow, O. G. Troyanskaya, H. Bussey, G. D. Bader, A.-C. Gingras, Q. D. Morris, P. M. Kim, C. A. Kaiser, C. L. Myers, B. J. Andrews, and C. Boone. The genetic landscape of a cell. *Science*, 327(5964):425–431, Jan. 2010.

[37] R. A. Craig and L. Liao. Phylogenetic tree information aids supervised learning for predicting protein-protein interaction based on distance matrices. *BMC Bioinformatics*, 8(1):6+, Jan. 2007.

[38] M. E. Cusick, N. Klitgord, M. Vidal, and D. E. Hill. Interactome: gateway into systems biology. *Human Molecular Genetics*, 14(suppl 2):R171–R181, Oct. 2005.

[39] C. O. Daub, R. Steuer, J. Selbig, and S. Kloska. Estimating mutual information using b-spline functions–an improved similarity measure for analysing gene expression data. *BMC Bioinformatics*, 5(1):118+, Aug. 2004.

[40] M. J. Davis, C. J. J. Shin, N. Jing, and M. A. Ragan. Rewiring the dynamic interactome. *Molecular BioSystems*, 8(8):2054–2066, Aug. 2012.

[41] U. de Lichtenberg, L. J. Jensen, S. Brunak, and P. Bork. Dynamic complex formation during the yeast cell cycle. *Science*, 307(5710):724–727, Feb. 2005.

[42] M. Deng, K. Zhang, S. Mehta, T. Chen, and F. Sun. Prediction of protein function using protein-protein interaction data. *Journal of Computational Biology*, 10(6):947–960, 2003.

[43] C. Dessimoz, N. Škunca, and P. D. Thomas. CAFA and the open world of protein function predictions. *Trends in Genetics*, 29(11):609–610, Nov. 2013.

[44] P. D'haeseleer, S. Liang, and R. Somogyi. Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, 16(8):707–726, Aug. 2000.

[45] J. Dong and S. Horvath. Understanding network concepts in modules. *BMC Systems Biology*, 1(1):24, 2007.

[46] 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, Dec. 1998.

[47] J. D. Ellis, M. Barrios-Rodiles, R. Çolak, M. Irimia, T. Kim, J. A. Calarco, X. Wang, Q. Pan, D. O'Hanlon, P. M. Kim, J. L. Wrana, and B. J. Blencowe. Tissue-Specific alternative splicing remodels Protein-Protein interaction networks. *Molecular Cell*, 46(6):884–892, June 2012.

[48] D. C. Ellwanger, J. F. Leonhardt, and H.-W. Mewes. Large-scale modeling of condition-specific gene regulatory networks by information integration and inference. *Nucleic Acids Research*, 42(21):000, Dec. 2014.

[49] A. J. Enright, I. Iliopoulos, N. C. Kyrpides, and C. A. Ouzounis. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402(6757):86–90, Nov. 1999.

[50] P. Erdös and A. Rényi. On random graphs, i. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.

[51] A. Fasano, I. Berti, T. Gerarduzzi, T. Not, R. B. Colletti, S. Drago, Y. Elitsur, P. H. Green, S. Guandalini, I. D. Hill, M. Pietzak, A. Ventura, M. Thorpe, D. Kryszak, F. Fornaroli, S. S. Wasserman, J. A. Murray, and K. Horvath. Prevalence of celiac disease in at-risk and not-at-risk groups in the united states: a large multicenter study. *Archives of Internal Medicine*, 163(3):286–292, Feb. 2003.

[52] L. Ferrer, J. M. Dale, and P. D. Karp. A systematic study of genome context methods: calibration, normalization and combination. *BMC Bioinformatics*, 11(1):493+, 2010.

[53] F. Fouss, A. Pirotte, J. M. Renders, and M. Saerens. Random-Walk computation of similarities between nodes of a graph with application to collaborative recommendation. *Knowledge and Data Engineering, IEEE Transactions on*, 19(3):355–369, Mar. 2007.

[54] C. C. Friedel and R. Zimmer. Toward the complete interactome. *Nature Biotechnology*, 24(6):614–615, June 2006.

[55] I. Gaillard, S. Rouquier, and D. Giorgi. Olfactory receptors. *Cellular and Molecular Life Sciences : CMLS*, 61(4):456–469, Feb. 2004.

[56] C. Gaiteri, Y. Ding, B. French, G. C. Tseng, and E. Sibille. Beyond modules and hubs: the potential of gene coexpression networks for investigating molecular mechanisms of complex brain disorders. *Genes, Brain, and Behavior*, 13(1):13–24, Jan. 2014.

[57] R. S. Galhardo, P. J. Hastings, and S. M. Rosenberg. Mutation as a stress response and the regulation of evolvability. *Critical Reviews in Biochemistry and Molecular Biology*, 42(5):399–435, Jan. 2007.

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

[59] A.-C. Gavin, K. Maeda, and S. Kühner. Recent advances in charting protein–protein interaction: mass spectrometry-based approaches. *Current Opinion in Biotechnology*, 22(1):42–49, Feb. 2011.

[60] A.-C. C. Gavin, P. Aloy, P. Grandi, R. Krause, M. Boesche, M. Marzioch, C. Rau, L. J. J. Jensen, S. Bastuck, B. Dümpelfeld, A. Edelmann, M.-A. A. Heurtier, V. Hoffman, C. Hoefert, K. Klein, M. Hudak, A.-M. M. Michon, M. Schelder, M. Schirle, M. Remor, T. Rudi, S. Hooper, A. Bauer, T. Bouwmeester, G. Casari, G. Drewes, G. Neubauer, J. M. Rick, B. Kuster, P. Bork, R. B. Russell, and G. Superti-Furga. Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440(7084):631–636, Mar. 2006.

[61] A.-C. C. Gavin, M. Bösche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. M. Rick, A.-M. M. Michon, C.-M. M. Cruciat, M. Remor, C. Höfert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M.-A. A. Heurtier, R. R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer, and G. Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, Jan. 2002.

[62] D. Gems and L. Partridge. Stress-response hormesis and aging:"that which does not kill us makes us stronger". *Cell Metabolism*, 7(3):200–203, 2008.

[63] P. Gerlee, L. Lizana, and K. Sneppen. Pathway identification by network pruning in the metabolic network of escherichia coli. *Bioinformatics*, 25(24):3282–3288, Dec. 2009.

[64] J. Gillis and P. Pavlidis. The impact of multifunctional genes on "guilt by association" analysis. *PLoS One*, 6(2):e17258+, Feb. 2011.

[65] J. Gillis and P. Pavlidis. "guilt by association" is the exception rather than the rule in gene networks. *PLoS Computational Biology*, 8(3):e1002444+, Mar. 2012.

[66] J. Gillis and P. Pavlidis. Assessing identity, redundancy and confounds in gene ontology annotations over time. *Bioinformatics*, 29(4):476–482, Feb. 2013.

[67] J. Gillis and P. Pavlidis. Characterizing the state of the art in the computational assignment of gene function: lessons from the first critical assessment of functional annotation (CAFA). *BMC Bioinformatics*, 14(Suppl 3):S15+, 2013.

[68] F. M. Giorgi, C. Del Fabbro, and F. Licausi. Comparative study of RNA-seq- and microarray-derived coexpression networks in arabidopsis thaliana. *Bioinformatics*, 29(6):717–724, Mar. 2013.

[69] A. Giuliani, S. Filippi, and M. Bertolaso. Why network approach can promote a new way of thinking in biology. *Frontiers in Genetics*, 5, 2014.

[70] L. Glass and S. A. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *Journal of Theoretical Biology*, 39(1):103–129, Apr. 1973.

[71] P. V. Gopalacharyulu, V. R. Velagapudi, E. Lindfors, E. Halperin, and M. Oresic. Dynamic network topology changes in functional modules predict responses to oxidative stress in yeast. *Molecular BioSystems*, 5(3):276–287, 2009.

[72] B. R. Graveley. Alternative splicing: increasing diversity in the proteomic world. *Trends in Genetics*, 17(2):100–107, Feb. 2001.

[73] R. Guimera and L. A. Nunes Amaral. Functional cartography of complex metabolic networks. *Nature*, 433(7028):895–900, Feb. 2005.

[74] U. Güldener, M. Münsterkötter, M. Oesterheld, P. Pagel, A. Ruepp, H.-W. W. Mewes, and V. Stümpflen. MPact: the MIPS protein interaction resource on yeast. *Nucleic Acids Research*, 34(Database issue):D436–D441, Jan. 2006.

[75] M. W. Hahn and A. D. Kern. Comparative genomics of centrality and essentiality in three eukaryotic Protein-Interaction networks. *Molecular Biology and Evolution*, 22(4):803–806, Apr. 2005.

[76] L. Hakes, J. W. Pinney, D. L. Robertson, and S. C. Lovell. Protein-protein interaction networks and biology—what's the connection? *Nature Biotechnology*, 26(1):69–72, Jan. 2008.

[77] A. Hamosh, A. F. Scott, J. Amberger, C. Bocchini, D. Valle, and V. A. McKusick. Online mendelian inheritance in man (OMIM), a knowledge-base of human genes and genetic disorders. *Nucleic Acids Research*, 30(1):52–55, Jan. 2002.

[78] J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. M. Walhout, M. E. Cusick, F. P. Roth, and M. Vidal. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6995):88–93, July 2004.

[79] J.-D. J. Han, D. Dupuy, N. Bertin, M. E. Cusick, and M. Vidal. Effect of sampling on topology predictions of protein-protein interaction networks. *Nature Biotechnology*, 23(7):839–844, July 2005.

[80] G. T. Hart, A. K. Ramani, and E. M. Marcotte. How complete are current yeast and human protein-interaction networks? *Genome Biology*, 7(11):120+, Dec. 2006.

[81] L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–C52, Dec. 1999.

[82] J.-K. Hériché, J. G. Lees, I. Morilla, T. Walter, B. Petrova, M. J. Roberti, M. J. Hossain, P. Adler, J. M. Fernández, M. Krallinger, C. H. Haering, J. Vilo, A. Valencia, J. A. Ranea, C. Orengo, and J. Ellenberg. Integration of biological data by kernels on graph nodes allows prediction of new genes involved in mitotic chromosome condensation. *Molecular Biology of the Cell*, 25(16):mbc.E13–04–0221–2536, June 2014.

[83] D. J. Higham, M. Rašajski, and N. Pržulj. Fitting a geometric graph to a protein–protein interaction network. *Bioinformatics*, 24(8):1093–1099, Apr. 2008.

[84] A. Hintze and C. Adami. Evolution of complex modular biological networks. *PLoS Computational Biology*, 4(2):e23+, Feb. 2008.

[85] H. Hishigaki, K. Nakai, T. Ono, A. Tanigami, and T. Takagi. Assessment of prediction accuracy of protein function from protein–protein interaction data. *Yeast*, 18(6):523–531, Apr. 2001.

[86] Y. Ho, A. Gruhler, A. Heilbut, G. D. Bader, L. Moore, S.-L. Adams, A. Millar, P. Taylor, K. Bennett, K. Boutilier, L. Yang, C. Wolting, I. Donaldson, S. Schandorff, J. Shewnarane, M. Vo, J. Taggart, M. Goudreault, B. Muskat, C. Alfarano, D. Dewar, Z. Lin, K. Michalickova, A. R. Willems,

H. Sassi, P. A. Nielsen, K. J. Rasmussen, J. R. Andersen, L. E. Johansen, L. H. Hansen, H. Jespersen, A. Podtelejnikov, E. Nielsen, J. Crawford, V. Poulsen, B. D. Sorensen, J. Matthiesen, R. C. Hendrickson, F. Gleeson, T. Pawson, M. F. Moran, D. Durocher, M. Mann, C. W. V. Hogue, D. Figeys, and M. Tyers. Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature*, 415(6868):180–183, Jan. 2002.

[87] R. Hoffmann and A. Valencia. Implementing the iHOP concept for navigation of biomedical literature. *Bioinformatics*, 21(suppl 2):ii252–ii258, Jan. 2005.

[88] P. Holme. Metabolic robustness and network modularity: a model study. *PLoS One*, 6(2):e16605+, Feb. 2011.

[89] P. Holme, B. J. J. Kim, C. N. N. Yoon, and S. K. K. Han. Attack vulnerability of complex networks. *Physical Review E*, 65(5 Pt 2), May 2002.

[90] N. Huang, I. Lee, E. M. Marcotte, and M. E. Hurles. Characterising and predicting haploinsufficiency in the human genome. *PLoS Genetics*, 6(10):e1001154+, Oct. 2010.

[91] L. A. Huber. Is proteomics heading in the wrong direction? *Nature Reviews Molecular Cell Biology*, 4(1):74–80, Jan. 2003.

[92] M. Huss and P. Holme. Currency and commodity metabolites: Their identification and relation to the modularity of metabolic networks. *IET Systems Biology*, 1(5):280–285, Mar. 2006.

[93] W. C. Hwang, A. Zhang, and M. Ramanathan. Identification of information Flow-Modulating drug targets: A novel bridging paradigm for drug discovery. *Clinical Pharmacology Therapeutics*, 84(5):563–572, July 2008.

[94] T. Ideker and N. J. Krogan. Differential network biology. *Molecular Systems Biology*, 8(1):565, 2012.

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

[96] I. Ispolatov, P. L. Krapivsky, and A. Yuryev. Duplication-divergence model of protein interaction network. *Physical Review E*, 71(6 Pt 1), June 2005.

[97] H. Ito, T. Oshiro, Y. Fujita, S. Kubota, C. Naito, H. Ohtsuka, H. Murakami, and H. Aiba. Pma1, a p-type proton ATPase, is a determinant of chronological life span in fission yeast. *Journal of Biological Chemistry*, 285(45):34616–34620, Nov. 2010.

[98] T. Ito, K. Tashiro, S. Muta, R. Ozawa, T. Chiba, M. Nishizawa, K. Yamamoto, S. Kuhara, and Y. Sakaki. Toward a protein-protein interaction map of the budding yeast: A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proceedings of the National Academy of Sciences*, 97(3):1143–1147, Feb. 2000.

[99] V. Janjić, R. Sharan, and N. Pržulj. Modelling the yeast interactome. *Scientific Reports*, 4, 2014.

[100] D. C. Jeffares, C. J. Penkett, and J. Bähler. Rapidly regulated genes are intron poor. *Trends in Genetics*, 24(8):375–378, Aug. 2008.

[101] L. J. Jensen, M. Kuhn, M. Stark, S. Chaffron, C. Creevey, J. Muller, T. Doerks, P. Julien, A. Roth, M. Simonovic, P. Bork, and C. von Mering. STRING 8–a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, 37(Database issue):D412–D416, Jan. 2009.

[102] H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, May 2001.

[103] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A. L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, Oct. 2000.

[104] H. Jiang, L. Xu, and Z. Gu. Growth of novel epistatic interactions by gene duplication. *Genome Biology and Evolution*, 3:295–301, Jan. 2011.

[105] M. Kanehisa and S. Goto. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, Jan. 2000.

[106] U. Karaoz, T. M. Murali, S. Letovsky, Y. Zheng, C. Ding, C. R. Cantor, and S. Kasif. Whole-genome annotation by using evidence integration in functional-linkage networks. *Proceedings of the National Academy of Sciences*, 101(9):2888–2893, Mar. 2004.

[107] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. *Nature reviews. Molecular cell biology*, 9(10):770–780, Oct. 2008.

[108] H. Keren, G. Lev-Maor, and G. Ast. Alternative splicing and evolution: diversification, exon definition and function. *Nature Reviews Genetics*, 11(5):345–355, May 2010.

[109] S. Kerrien, Y. Alam-Faruque, B. Aranda, I. Bancarz, A. Bridge, C. Derow, E. Dimmer, M. Feuermann, A. Friedrichsen, R. Huntley, C. Kohler, J. Khadake, C. Leroy, A. Liban, C. Lieftink, L. Montecchi-Palazzi, S. Orchard, J. Risse, K. Robbe, B. Roechert, D. Thorneycroft, Y. Zhang, R. Apweiler, and H. Hermjakob. IntAct–open source resource for molecular interaction data. *Nucleic Acids Research*, 35(Database issue):D561–D565, Jan. 2007.

[110] R. Khanin and E. Wit. How scale-free are biological networks. *Journal of Computational Biology*, 13(3):810–818, Apr. 2006.

[111] E. Khurana, Y. Fu, J. Chen, and M. Gerstein. Interpretation of genomic variants using a unified biological network approach. *PLoS Computational Biology*, 9(3):e1002886+, Mar. 2013.

[112] E. Khurana, Y. Fu, V. Colonna, X. J. Mu, H. M. Kang, T. Lappalainen, A. Sboner, L. Lochovsky, J. Chen, A. Harmanci, J. Das, A. Abyzov, S. Balasubramanian, K. Beal, D. Chakravarty, D. Challis, Y. Chen, D. Clarke, L. Clarke, F. Cunningham, U. S. Evani, P. Flicek, R. Fragoza, E. Garrison, R. Gibbs, Z. H. Gümüş, J. Herrero, N. Kitabayashi, Y. Kong, K. Lage, V. Liluashvili, S. M. Lipkin, D. G. MacArthur, G. Marth, D. Muzny, T. H. Pers, G. R. S. Ritchie, J. A. Rosenfeld, C. Sisu, X. Wei, M. Wilson, Y. Xue,

F. Yu, 1000 Genomes Project Consortium, E. T. Dermitzakis, H. Yu, M. A. Rubin, C. Tyler-Smith, and M. Gerstein. Integrative annotation of variants from 1092 humans: Application to cancer genomics. *Science*, 342(6154):1235587+, Oct. 2013.

[113] P.-J. Kim, D.-Y. Lee, T. Y. Kim, K. H. Lee, H. Jeong, S. Y. Lee, and S. Park. Metabolite essentiality elucidates robustness of escherichia coli metabolism. *Proceedings of the National Academy of Sciences*, 104(34):13638–13642, Aug. 2007.

[114] W. K. K. Kim, J. Park, and J. K. K. Suh. Large scale statistical prediction of protein-protein interaction by potentially interacting domain (PID) pair. *Genome informatics. International Conference on Genome Informatics*, 13:42–50, 2002.

[115] H. Kitano. Biological robustness. *Nature Reviews Genetics*, 5(11):826–837, Nov. 2004.

[116] H. Kitano. A robustness-based approach to systems-oriented drug design. *Nature Reviews Drug Discovery*, 6(3):202–210, Mar. 2007.

[117] E. Klipp, B. Nordlander, R. Kruger, P. Gennemark, and S. Hohmann. Integrative model of the response of yeast to osmotic shock. *Nature Biotechnology*, 23(8):975–982, July 2005.

[118] M. Kollmann, L. Lovdok, K. Bartholome, J. Timmer, and V. Sourjik. Design principles of a bacterial signalling network. *Nature*, 438(7067):504–507, Nov. 2005.

[119] K. Komurov and M. White. Revealing static and dynamic modular architecture of the eukaryotic protein interaction network. *Molecular Systems Biology*, 3(1), Apr. 2007.

[120] R. I. Kondor and J. D. Lafferty. *Diffusion Kernels on Graphs and Other Discrete Input Spaces*. ICML '02. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[121] Y. A. I. Kourmpetis, A. D. J. van Dijk, M. C. A. M. Bink, R. C. H. J. van Ham, and C. J. F. ter Braak. Bayesian markov random field analysis for protein function prediction based on network data. *PLoS One*, 5(2):e9293+, Feb. 2010.

[122] I. A. Kovács, R. Palotai, M. S. Szalay, and P. Csermely. Community landscapes: An integrative approach to determine overlapping network module hierarchy, identify key nodes and predict network dynamics. *PLoS One*, 5(9):e12528+, Sept. 2010.

[123] N. J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, A. Ignatchenko, J. Li, S. Pu, N. Datta, A. P. Tikuisis, T. Punna, J. M. Peregrín-Alvarez, M. Shales, X. Zhang, M. Davey, M. D. Robinson, A. Paccanaro, J. E. Bray, A. Sheung, B. Beattie, D. P. Richards, V. Canadien, A. Lalev, F. Mena, P. Wong, A. Starostine, M. M. Canete, J. Vlasblom, S. Wu, C. Orsi, S. R. Collins, S. Chandran, R. Haw, J. J. Rilstone, K. Gandi, N. J. Thompson, G. Musso, P. St Onge, S. Ghanny, M. H. Lam, G. Butland, A. M. Altaf-Ul, S. Kanaya, A. Shilatifard, E. O'Shea, J. S. Weissman, C. J. Ingles, T. R.

Hughes, J. Parkinson, M. Gerstein, S. J. Wodak, A. Emili, and J. F. Greenblatt. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 440(7084):637–643, Mar. 2006.

[124] L. Kruglyak and D. A. Nickerson. Variation is the spice of life. *Nature Genetics*, 27(3):234–236, Mar. 2001.

[125] D. Kültz. Molecular and evolutionary basis of the cellular stress response. *Annual Review of Physiology*, 67(1):225–257, 2005.

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

[127] G. R. Lanckriet, M. Deng, N. Cristianini, M. I. Jordan, and W. S. Noble. Kernel-based data fusion and its application to protein function prediction in yeast. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pages 300–311, 2004.

[128] M. Lappe and L. Holm. Unraveling protein interaction networks with near-optimal efficiency. *Nature Biotechnology*, 22(1):98–103, Dec. 2003.

[129] K. A. Le Cao, S. Boitard, and P. Besse. Sparse PLS discriminant analysis: biologically relevant feature selection and graphical displays for multiclass problems. *BMC Bioinformatics*, 12(1):253+, 2011.

[130] H. Lee, Z. Tu, M. Deng, F. Sun, and T. Chen. Diffusion kernel-based logistic regression models for protein function prediction. *Omics: A Journal of Integrative Biology*, 10(1):40–55, 2006.

[131] J. G. Lees, J. K. Heriche, I. Morilla, J. A. Ranea, and C. A. Orengo. Systematic computational prediction of protein interaction networks. *Physical Biology*, 8(3):035008+, June 2011.

[132] B. Lehner. Genes confer similar robustness to environmental, stochastic, and genetic perturbations in yeast. *PLoS One*, 5(2):e9035+, Feb. 2010.

[133] S. Lehtinen, F. X. X. Marsellach, S. Codlin, A. Schmidt, M. Clément-Ziza, A. Beyer, J. Bähler, C. Orengo, and V. Pancaldi. Stress induces remodelling of yeast interaction and co-expression networks. *Molecular BioSystems*, 9(7):1697–1707, July 2013.

[134] S. Letovsky and S. Kasif. Predicting protein function from protein/protein interaction data: a probabilistic approach. *Bioinformatics*, 19 Suppl 1(suppl 1):i197–i204, July 2003.

[135] J. Li, B. Horstman, and Y. Chen. Detecting epistatic effects in association studies at a genomic level based on an ensemble approach. *Bioinformatics*, 27(13):i222–i229, July 2011.

[136] L. Licata, L. Briganti, D. Peluso, L. Perfetto, M. Iannuccelli, E. Galeota, F. Sacco, A. Palma, A. P. P. Nardozza, E. Santonico, L. Castagnoli, and G. Cesareni. MINT, the molecular interaction database: 2012 update. *Nucleic Acids Research*, 40(Database issue):D857–D861, Jan. 2012.

[137] G. Lima-Mendez and J. van Helden. The powerful law of the power law and other myths in network biology. *Molecular BioSystems*, 5(12):1482–1493, Dec. 2009.

[138] A. Lin, R. T. Wang, S. Ahn, C. C. Park, and D. J. Smith. A genome-wide map of human genetic interactions inferred from radiation hybrid genotypes. *Genome Research*, 20(8):1122–1132, Aug. 2010.

[139] T.-W. W. Lin, J.-W. W. Wu, and D. T.-H. T. Chang. Combining phylogenetic profiling-based and machine learning-based techniques to predict functional related proteins. *PLoS One*, 8(9), 2013.

[140] L. López-Maury, S. Marguerat, and J. Bähler. Tuning gene expression to changing environments: from rapid responses to evolutionary adaptation. *Nature Reviews Genetics*, 9(8):583–593, Aug. 2008.

[141] N. M. Luscombe, M. Madan Babu, H. Yu, M. Snyder, S. A. Teichmann, and M. Gerstein. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431(7006):308–312, Sept. 2004.

[142] H. Ma and A.-P. Zeng. Reconstruction of metabolic networks from genome data and analysis of their global structure for various organisms. *Bioinformatics*, 19(2):270–277, Jan. 2003.

[143] D. G. MacArthur, S. Balasubramanian, A. Frankish, N. Huang, J. Morris, K. Walter, L. Jostins, L. Habegger, J. K. Pickrell, S. B. Montgomery, C. A. Albers, Z. D. Zhang, D. F. Conrad, G. Lunter, H. Zheng, Q. Ayub, M. A. DePristo, E. Banks, M. Hu, R. E. Handsaker, J. A. Rosenfeld, M. Fromer, M. Jin, X. J. J. Mu, E. Khurana, K. Ye, M. Kay, G. I. I. Saunders, M.-M. M. Suner, T. Hunt, I. H. Barnes, C. Amid, D. R. Carvalho-Silva, A. H. Bignell, C. Snow, B. Yngvadottir, S. Bumpstead, D. N. Cooper, Y. Xue, I. G. G. Romero, 1000 Genomes Project Consortium, J. Wang, Y. Li, R. A. Gibbs, S. A. McCarroll, E. T. Dermitzakis, J. K. Pritchard, J. C. Barrett, J. Harrow, M. E. Hurles, M. B. Gerstein, and C. Tyler-Smith. A systematic survey of loss-of-function variants in human protein-coding genes. *Science*, 335(6070):823–828, Feb. 2012.

[144] D. G. MacArthur and C. Tyler-Smith. Loss-of-function variants in the genomes of healthy humans. *Human Molecular Genetics*, 19(R2):R125–R130, Oct. 2010.

[145] R. Manne. Analysis of two partial-least-squares algorithms for multivariate calibration. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):187–197, Aug. 1987.

[146] C. D. Manning, P. Raghavan, H. Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.

[147] S. Marguerat, A. Schmidt, S. Codlin, W. Chen, R. Aebersold, and J. Bähler. Quantitative analysis of fission yeast transcriptomes and proteomes in proliferating and quiescent cells. *Cell*, 151(3):671–683, Oct. 2012.

[148] F. Markowetz and R. Spang. Inferring cellular networks–a review. *BMC Bioinformatics*, 8 Suppl 6(Suppl 6):S5+, 2007.

[149] A. J. Matlin, F. Clark, and C. W. J. Smith. Understanding alternative splicing: towards a cellular code. *Nature Reviews Molecular Cell Biology*, 6(5):386–398, May 2005.

[150] L. R. Matthews, P. Vaglio, J. Reboul, H. Ge, B. P. Davis, J. Garrels, S. Vincent, and M. Vidal. Identification of potential interaction networks using sequence-based searches for conserved protein-protein interactions or "interologs". *Genome Research*, 11(12):2120–2126, Dec. 2001.

[151] M. L. Metzker. Sequencing technologies [mdash] the next generation. *Nature Reviews Genetics*, 11(1):31–46, Jan. 2010.

[152] A. Mihalik and P. Csermely. Heat shock partially dissociates the overlapping modules of the yeast Protein-Protein interaction network: A systems level model of adaptation. *PLoS Computational Biology*, 7(10):e1002187+, Oct. 2011.

[153] B. Modrek and C. Lee. A genomic view of alternative splicing. *Nature Genetics*, 30(1):13–19, Jan. 2002.

[154] S. J. Moodie, P. J. Norman, A. L. King, J. S. Fraser, D. Curtis, H. J. Ellis, R. W. Vaughan, and P. J. Ciclitira. Analysis of candidate genes on chromosome 19 in coeliac disease: an association study of the KIR and LILR gene clusters. *European Journal of Immunogenetics*, 29(4):287–291, Aug. 2002.

[155] S. Mostafavi, D. Ray, D. Warde-Farley, C. Grouios, and Q. Morris. GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biology*, 9 Suppl 1(Suppl 1):S4+, 2008.

[156] P. Myllymäki, T. Silander, H. Tirri, and P. Uronen. B-course: A web-based tool for bayesian and causal data analysis. *International Journal on Artificial Intelligence Tools*, 11(03):369–387, Sept. 2002.

[157] E. Nabieva, K. Jim, A. Agarwal, B. Chazelle, and M. Singh. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics*, 21(suppl 1):i302–i310, June 2005.

[158] M. E. Newman. Models of the small world. *Journal of Statistical Physics*, 101(3-4):819–841, 2000.

[159] M. E. Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[160] M. E. Newman. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330, 2004.

[161] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3):036104+, Sept. 2006.

[162] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, June 2006.

[163] T. Ni, K. Tu, Z. Wang, S. Song, H. Wu, B. Xie, K. C. Scott, S. I. Grewal, Y. Gao, and J. Zhu. The prevalence and regulation of antisense transcripts in schizosaccharomyces pombe. *PLoS One*, 5(12):e15271+, Dec. 2010.

[164] A. Özgür, T. Vu, G. Erkan, and D. R. Radev. Identifying gene-disease associations using centrality on a literature mined gene-interaction network. *Bioinformatics*, 24(13):i277–i285, July 2008.

[165] P. Pagel, S. Kovac, M. Oesterheld, B. Brauner, I. Dunger-Kaltenbach, G. Frishman, C. Montrone, P. Mark, V. Stümpflen, H.-W. Mewes, A. Ruepp, and D. Frishman. The MIPS mammalian protein–protein interaction database. *Bioinformatics*, 21(6):832–834, Mar. 2005.

[166] C. Pal, B. Papp, and L. D. Hurst. Genomic function (communication arising): Rate of evolution and gene dispensability. *Nature*, 421(6922):496–497, Jan. 2003.

[167] M. C. Palumbo, A. Colosimo, A. Giuliani, and L. Farina. Functional essentiality from topology features in metabolic networks: A case study in yeast. *FEBS Letters*, 579(21):4642–4646, Aug. 2005.

[168] V. Pancaldi, O. S. Saraç, C. Rallis, J. R. McLean, M. Převorovský, K. Gould, A. Beyer, and J. Bähler. Predicting the fission yeast protein interaction network. *G3: Genes—Genomes—Genetics*, 2(4):453–467, Apr. 2012.

[169] V. Pancaldi, F. Schubert, and J. Bähler. Meta-analysis of genome regulation and expression variability across hundreds of environmental and genetic perturbations in fission yeast. *Molecular BioSystems*, 6(3):543–552, Mar. 2010.

[170] J. A. Papin, T. Hunter, B. O. Palsson, and S. Subramaniam. Reconstruction of cellular signalling networks and analysis of their properties. *Nature Reviews Molecular Cell Biology*, 6(2):99–111, Feb. 2005.

[171] M. Pardo, B. Lang, L. Yu, H. Prosser, A. Bradley, M. M. Babu, and J. Choudhary. An expanded oct4 interaction network: implications for stem cell biology, development, and disease. *Cell Stem Cell*, 6(4):382–395, Apr. 2010.

[172] K. Park and D. Kim. Localized network centrality and essentiality in the yeast–protein interaction network. *Proteomics*, 9(22):5143–5154, Nov. 2009.

[173] A. S. Parmar, N. Alakulppi, P. Paavola-Sakki, K. Kurppa, L. Halme, M. Färkkilä, U. Turunen, M. Lappalainen, K. Kontula, K. Kaukinen, M. Mäki, K. Lindfors, J. Partanen, P. Sistonen, J. Mättö, P. Wacklin, P. Saavalainen, and E. Einarsdottir. Association study of FUT2 (rs601338) with celiac disease and inflammatory bowel disease in the finnish population. *Tissue Antigens*, 80(6):488–493, Dec. 2012.

[174] F. Pazos, M. Helmer-Citterich, G. Ausiello, and A. Valencia. Correlated mutations contain information about protein-protein interaction. *Journal of Molecular Biology*, 271(4):511–523, Aug. 1997.

[175] L. Peña Castillo, M. Tasan, C. L. Myers, H. Lee, T. Joshi, C. Zhang, Y. Guan, M. Leone, A. Pagnani, W. K. K. Kim, C. Krumpelman, W. Tian, G. Obozinski, Y. Qi, S. Mostafavi, G. N. N. Lin, G. F. Berriz, F. D. Gibbons, G. Lanckriet, J. Qiu, C. Grant, Z. Barutcuoglu, D. P. Hill, D. Warde-Farley, C. Grouios, D. Ray, J. A. Blake, M. Deng, M. I. Jordan, W. S. Noble, Q. Morris, J. Klein-Seetharaman, Z. Bar-Joseph, T. Chen, F. Sun, O. G. Troyanskaya, E. M. Marcotte, D. Xu, T. R. Hughes, and F. P. Roth. A critical assessment of mus musculus gene function prediction using integrated genomic evidence. *Genome Biology*, 9 Suppl 1(Suppl 1):S2+, 2008.

[176] M. Pellegrini, E. M. Marcotte, M. J. Thompson, D. Eisenberg, and T. O. Yeates. Assigning protein functions by comparative genome analysis: Protein phylogenetic profiles. *Proceedings of the National Academy of Sciences*, 96(8):4285–4288, Apr. 1999.

[177] J. Peng, P. Wang, N. Zhou, and J. Zhu. Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, 104(486):735–746, June 2009.

[178] S. Peri, J. D. Navarro, R. Amanchy, T. Z. Kristiansen, C. K. Jonnalagadda, V. Surendranath, V. Niranjan, B. Muthusamy, T. K. B. Gandhi, M. Gronborg, N. Ibarrola, N. Deshpande, K. Shanker, H. N. Shivashankar, B. P. Rashmi, M. A. Ramya, Z. Zhao, K. N. Chandrika, N. Padma, H. C. Harsha, A. J. Yatish, M. P. Kavitha, M. Menezes, D. R. Choudhury, S. Suresh, N. Ghosh, R. Saravana, S. Chandran, S. Krishna, M. Joy, S. K. Anand, V. Madavan, A. Joseph, G. W. Wong, W. P. Schiemann, S. N. Constantinescu, L. Huang, R. Khosravi-Far, H. Steen, M. Tewari, S. Ghaffari, G. C. Blobe, C. V. Dang, J. G. N. Garcia, J. Pevsner, O. N. Jensen, P. Roepstorff, K. S. Deshpande, A. M. Chinnaiyan, A. Hamosh, A. Chakravarti, and A. Pandey. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome Research*, 13(10):2363–2371, Oct. 2003.

[179] S. Persson, H. Wei, J. Milne, G. P. Page, and C. R. Somerville. Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets. *Proceedings of the National Academy of Sciences*, 102(24):8633–8638, June 2005.

[180] C. Pesquita, D. Faria, A. O. Falcão, P. Lord, and F. M. Couto. Semantic similarity in biomedical ontologies. *PLoS Computational Biology*, 5(7):e1000443+, July 2009.

[181] M. Pigliucci. Is evolvability evolvable? *Nature Reviews Genetics*, 9(1):75–82, Jan. 2008.

[182] N. D. Price, J. A. Papin, C. H. Schilling, and B. O. Palsson. Genome-scale microbial in silico models: the constraints-based approach. *Trends in Biotechnology*, 21(4):162–169, Apr. 2003.

[183] N. Pržulj, D. G. Corneil, and I. Jurisica. Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, Dec. 2004.

[184] N. Pržulj, O. Kuchaiev, A. Stevanovic, and W. Hayes. Geometric evolutionary dynamics of protein interaction networks. *Biocomputing*, pages 178–189, 2010.

[185] Y. Qi, Z. Bar-Joseph, and J. Klein-Seetharaman. Evaluation of different biological data and computational classification methods for use in protein interaction prediction. *Proteins*, 63(3):490–500, May 2006.

[186] H. Qiu and E. R. Hancock. Clustering and embedding using commute times. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(11):1873–1890, Nov. 2007.

[187] P. Radivojac, W. T. Clark, T. R. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, A. Ben-Hur, G. Pandey, J. M. Yunes, A. S. Talwalkar, S. Repo, M. L. Souza, D. Piovesan, R. Casadio, Z. Wang, J. Cheng, H. Fang, J. Gough, P. Koskinen, P. Törönen, J. Nokso-Koivisto, L. Holm, D. Cozzetto, D. W. Buchan, K. Bryson, D. T. Jones, B. Limaye, H. Inamdar, A. Datta, S. K. Manjari, R. Joshi, M. Chitale, D. Kihara, A. M. Lisewski, S. Erdin, E. Venner, O. Lichtarge, R. Rentzsch, H. Yang, A. E. Romero, P. Bhat, A. Paccanaro, T. Hamp, R. Kaßner, S. Seemayer, E. Vicedo, C. Schaefer, D. Achten, F. Auer, A. Boehm, T. Braun, M. Hecht, M. Heron, P. Hönigschmid, T. A. Hopf, S. Kaufmann, M. Kiening, D. Krompass, C. Landerer, Y. Mahlich, M. Roos, J. Björne, T. Salakoski, A. Wong, H. Shatkay, F. Gatzmann, I. Sommer, M. N. Wass, M. J. Sternberg, N. Škunca, F. Supek, M. Bošnjak, P. Panov, S. Džeroski, T. Šmuc, Y. A. Kourmpetis, A. D. van Dijk, C. J. ter Braak, Y. Zhou, Q. Gong, X. Dong, W. Tian, M. Falda, P. Fontana, E. Lavezzo, B. Di Camillo, S. Toppo, L. Lan, N. Djuric, Y. Guo, S. Vucetic, A. Bairoch, M. Linial, P. C. Babbitt, S. E. Brenner, C. Orengo, B. Rost, S. D. Mooney, and I. Friedberg. A large-scale evaluation of computational protein function prediction. *Nature Methods*, 10(3):221–227, Mar. 2013.

[188] S. V. Rajagopala, K. T. Hughes, and P. Uetz. Benchmarking yeast two-hybrid systems using the interactions of bacterial motility proteins. *Proteomics*, 9(23):5296–5302, 2009.

[189] K. Raman, N. Damaraju, and G. K. Joshi. The organisational structure of protein networks: revisiting the centrality–lethality hypothesis. *Systems and Synthetic Biology*, 8(1):73–81, 2014.

[190] S. Rännar, F. Lindgren, P. Geladi, and S. Wold. A PLS kernel algorithm for data sets with many variables and fewer objects. part 1: Theory and algorithm. *Journal of Chemometrics*, 8(2):111–125, Mar. 1994.

[191] V. S. Rao, K. Srinivas, G. N. Sujini, and S. N. Kumar. Protein-protein interaction detection: methods and analysis. *International Journal of Proteomics*, 2014, 2014.

[192] E. Ravasz. Detecting hierarchical modularity in biological networks. *Methods in Molecular Biology*, 541:145–160, 2009.

[193] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabási. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551–1555, Aug. 2002.

[194] S. Razick, G. Magklaras, and I. M. Donaldson. iRefIndex: a consolidated protein interaction database with provenance. *BMC Bioinformatics*, 9(1):405+, 2008.

[195] D. E. Reich, S. B. Gabriel, and D. Altshuler. Quality and completeness of SNP databases. *Nature Genetics*, 33(4):457–458, Mar. 2003.

[196] O. Resendis-Antonio, J. A. Freyre-González, R. Menchaca-Méndez, R. M. Gutiérrez-Ríos, A. Martínez-Antonio, C. Avila-Sánchez, and J. Collado-Vides. Modular analysis of the transcriptional regulatory network of e. coli. *Trends in Genetics*, 21(1):16–20, Jan. 2005.

[197] H. Richard, M. H. Schulz, M. Sultan, A. Nürnberger, S. Schrinner, D. Balzereit, E. Dagand, A. Rasche, H. Lehrach, M. Vingron, S. A. Haas, and M.-L. Yaspo. Prediction of alternative isoforms from exon expression levels in RNA-seq experiments. *Nucleic Acids Research*, 38(10):e112, June 2010.

[198] G. Rigaut, A. Shevchenko, B. Rutz, M. Wilm, M. Mann, and B. Séraphin. A generic protein purification method for protein complex characterization and proteome exploration. *Nature Biotechnology*, 17(10):1030–1032, Oct. 1999.

[199] G. D. Rio, D. Koschutzki, and G. Coello. How to identify essential genes from molecular networks? *BMC Systems Biology*, 3(1):102+, 2009.

[200] A. Roberts, H. Pimentel, C. Trapnell, and L. Pachter. Identification of novel transcripts in annotated genomes using RNA-seq. *Bioinformatics*, 27(17):2325–2329, Sept. 2011.

[201] M. F. Rogers and A. Ben-Hur. The use of gene ontology evidence codes in preventing classifier assessment bias. *Bioinformatics*, 25(9):1173–1177, May 2009.

[202] A. E. Roux, A. Quissac, P. Chartrand, G. Ferbeyre, and L. A. Rokeach. Regulation of chronological aging in schizosaccharomyces pombe by the protein kinases pka1 and sck2. *Aging Cell*, 5(4):345–357, Aug. 2006.

[203] A. Ruepp, B. Brauner, I. Dunger-Kaltenbach, G. Frishman, C. Montrone, M. Stransky, B. Waegele, T. Schmidt, O. N. N. Doudieu, V. Stümpflen, and H. W. Mewes. CORUM: the comprehensive resource of mammalian protein complexes. *Nucleic Acids Research*, 36(Database issue):D646–D650, Jan. 2008.

[204] H. Saito and F. Posas. Response to hyperosmotic stress. *Genetics*, 192(2):289–318, Oct. 2012.

[205] L. Salwinski, C. S. Miller, A. J. Smith, F. K. Pettit, J. U. Bowie, and D. Eisenberg. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, 32(Database issue):D449–D451, Jan. 2004.

[206] T. Sato, Y. Yamanishi, M. Kanehisa, and H. Toh. The inference of protein-protein interactions by co-evolutionary analysis is improved by excluding the information about the phylogenetic relationships. *Bioinformatics*, 21(17):3482–3489, Sept. 2005.

[207] U. Sauer. High-throughput phenomics: experimental methods for mapping fluxomes. *Current Opinion in Biotechnology*, 15(1):58–63, Feb. 2004.

[208] J. Schäfer and K. Strimmer. An empirical bayes approach to inferring large-scale gene association networks. *Bioinformatics*, 21(6):754–764, Mar. 2005.

[209] E. E. Schmidt, O. Pelz, S. Buhlmann, G. Kerr, T. Horn, and M. Boutros. GenomeRNAi: a database for cell-based and in vivo RNAi phenotypes, 2013 update. *Nucleic Acids Research*, 41(D1):D1021–D1026, Jan. 2013.

[210] S. Schuster, T. Pfeiffer, F. Moldenhauer, I. Koch, and T. Dandekar. Exploring the pathway structure of metabolism: decomposition into subnetworks and application to mycoplasma pneumoniae. *Bioinformatics*, 18(2):351–361, Feb. 2002.

[211] A. S. Schwartz, J. Yu, K. R. Gardenour, R. L. Finley, and T. Ideker. Cost-effective strategies for completing the interactome. *Nature Methods*, 6(1):55–61, Jan. 2009.

[212] B. Schwikowski, P. Uetz, and S. Fields. A network of protein-protein interactions in yeast. *Nature Biotechnology*, 18(12):1257–1261, Dec. 2000.

[213] J. Shawe-Taylor. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[214] T. Sideri, C. Rallis, D. A. Bitton, B. M. Lages, F. Suo, M. Rodríguez-López, L.-L. Du, and J. Bähler. Parallel profiling of fission yeast deletion mutants for proliferation and for lifespan during long-term quiescence. *G3: Genes— Genomes— Genetics*, 5(1):145–155, 2015.

[215] R. Solomonoff and A. Rapoport. Connectivity of random nets. *Bulletin of Mathematical Biology*, 13(2):107–117, June 1951.

[216] L. Song, P. Langfelder, and S. Horvath. Comparison of co-expression measures: mutual information, correlation, and model based indices. *BMC Bioinformatics*, 13(1):328+, Dec. 2012.

[217] C. Stark, B.-J. J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34(Database issue):D535–D539, Jan. 2006.

[218] R. Steuer. Review: On the analysis and interpretation of correlations in metabolomic data. *Briefings in Bioinformatics*, 7(2):151–158, June 2006.

[219] S. H. Strogatz. Exploring complex networks. *Nature*, 410(6825):268–276, Mar. 2001.

[220] M. P. H. Stumpf and M. A. Porter. Critical truths about power laws. *Science*, 335(6069):665–666, Feb. 2012.

[221] M. P. H. Stumpf, T. Thorne, E. de Silva, R. Stewart, H. J. An, M. Lappe, and C. Wiuf. Estimating the size of the human interactome. *Proceedings of the National Academy of Sciences*, 105(19):6959–6964, May 2008.

[222] M. P. H. Stumpf, C. Wiuf, and R. M. May. Subnets of scale-free networks are not scale-free: Sampling properties of networks. *Proceedings of the National Academy of Sciences*, 102(12):4221–4224, Mar. 2005.

[223] M. Szalay-Bekő, R. Palotai, B. Szappanos, I. A. Kovács, B. Papp, and P. Csermely. ModuLand plug-in for cytoscape: determination of hierarchical layers of overlapping network modules and community centrality. *Bioinformatics*, 28(16):2202–2204, Aug. 2012.

[224] K. Tarassov, V. Messier, C. R. Landry, S. Radinovic, M. M. Serna Molina, I. Shames, Y. Malitskaya, J. Vogel, H. Bussey, and S. W. Michnick. An in vivo map of the yeast protein interactome. *Science*, 320(5882):1465–1470, June 2008.

[225] S. Tarazona, F. García-Alcalde, J. Dopazo, A. Ferrer, and A. Conesa. Differential expression in rna-seq: a matter of depth. *Genome research*, 21(12):2213–2223, 2011.

[226] I. Thiele, N. Swainston, R. M. T. Fleming, A. Hoppe, S. Sahoo, M. K. Aurich, H. Haraldsdottir, M. L. Mo, O. Rolfsson, M. D. Stobbe, S. G. Thorleifsson, R. Agren, C. Bolling, S. Bordel, A. K. Chavali, P. Dobson, W. B. Dunn, L. Endler, D. Hala, M. Hucka, D. Hull, D. Jameson, N. Jamshidi, J. J. Jonsson, N. Juty, S. Keating, I. Nookaew, N. Le Novere, N. Malys, A. Mazein, J. A. Papin, N. D. Price, E. Selkov, M. I. Sigurdsson, E. Simeonidis, N. Sonnenschein, K. Smallbone, A. Sorokin, J. H. G. M. van Beek, D. Weichart, I. Goryanin, J. Nielsen, H. V. Westerhoff, D. B. Kell, P. Mendes, and B. O. Palsson. A community-driven global reconstruction of human metabolism. *Nature Biotechnology*, 31(5):419–425, Mar. 2013.

[227] I. Tirosh, A. Weinberger, M. Carmi, and N. Barkai. A genetic signature of interspecies variations in gene expression. *Nature Genetics*, 38(7):830–834, June 2006.

[228] A. H. Y. Tong, G. Lesage, G. D. Bader, H. Ding, H. Xu, X. Xin, J. Young, G. F. Berriz, R. L. Brost, M. Chang, Y. Chen, X. Cheng, G. Chua, H. Friesen, D. S. Goldberg, J. Haynes, C. Humphries, G. He, S. Hussein, L. Ke, N. Krogan, Z. Li, J. N. Levinson, H. Lu, P. Ménard, C. Munyana, A. B. Parsons, O. Ryan, R. Tonikian, T. Roberts, A.-M. Sdicu, J. Shapiro, B. Sheikh, B. Suter, S. L. Wong, L. V. Zhang, H. Zhu, C. G. Burd, S. Munro, C. Sander, J. Rine, J. Greenblatt, M. Peter, A. Bretscher, G. Bell, F. P. Roth, G. W. Brown, B. Andrews, H. Bussey, and C. Boone. Global mapping of the yeast genetic interaction network. *Science*, 303(5659):808–813, Feb. 2004.

[229] S. Trajanovski, J. Martín-Hernández, W. Winterbach, and P. Van Mieghem. Robustness envelopes of networks. *Journal of Complex Networks*, 1(1):44–62, June 2013.

[230] T.-D. Tran and Y.-K. Kwon. The relationship between modularity and robustness in signalling networks. *Journal of The Royal Society Interface*, 10(88):20130771+, Nov. 2013.

[231] K. Tsuda, H. Shin, and B. Schölkopf. Fast protein classification with multiple networks. *Bioinformatics*, 21(suppl 2):ii59–ii65, Jan. 2005.

[232] N. Tuncbag, A. Gursoy, E. Guney, R. Nussinov, and O. Keskin. Architectures and functional coverage of protein-protein interfaces. *Journal of Molecular Biology*, 381(3):785–802, Sept. 2008.

[233] F. Vaggi, J. Dodgson, A. Bajpai, A. Chessel, F. Jordán, M. Sato, R. E. Carazo-Salas, and A. Csikász-Nagy. Linkers of cell polarity and cell cycle regulation in the fission yeast protein interaction network. *PLoS Computational Biology*, 8(10):e1002732+, Oct. 2012.

[234] D. L. van den Berg, T. Snoek, N. P. Mullin, A. Yates, K. Bezstarosti, J. Demmers, I. Chambers, and R. A. Poot. An oct4-centered protein interaction network in embryonic stem cells. *Cell Stem Cell*, 6(4):369–381, Apr. 2010.

[235] A. Vazquez, A. Flammini, A. Maritan, and A. Vespignani. Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, 21(6):697–700, June 2003.

[236] K. Venkatesan, J.-F. Rual, A. Vazquez, U. Stelzl, I. Lemmens, T. Hirozane-Kishikawa, T. Hao, M. Zenkner, X. Xin, K.-I. Goh, M. A. Yildirim, N. Simonis, K. Heinzmann, F. Gebreab, J. M. Sahalie, S. Cevik, C. Simon, A.-S. de Smet, E. Dann, A. Smolyar, A. Vinayagam, H. Yu, D. Szeto, H. Borick, A. Dricot, N. Klitgord, R. R. Murray, C. Lin, M. Lalowski, J. Timm, K. Rau, C. Boone, P. Braun, M. E. Cusick, F. P. Roth, D. E. Hill, J. Tavernier, E. E. Wanker, A.-L. Barabasi, and M. Vidal. An empirical framework for binary interactome mapping. *Nature Methods*, 6(1):83–90, Jan. 2009.

[237] M. Vidal and S. Fields. The yeast two-hybrid assay: still finding connections after 25 years. *Nature Methods*, 11(12):1203–1206, 2014.

[238] I. Voineagu, X. Wang, P. Johnston, J. K. Lowe, Y. Tian, S. Horvath, J. Mill, R. M. Cantor, B. J. Blencowe, and D. H. Geschwind. Transcriptomic analysis of autistic brain reveals convergent molecular pathology. *Nature*, 474(7351):380–384, June 2011.

[239] C. von Mering, R. Krause, B. Snel, M. Cornell, S. G. Oliver, S. Fields, and P. Bork. Comparative assessment of large-scale data sets of protein-protein interactions. *Nature*, 417(6887):399–403, May 2002.

[240] A. Wagner and D. A. Fell. The small world inside large metabolic networks. *Proceedings of the Royal Society of London B: Biological Sciences*, 268(1478):1803–1810, 2001.

[241] B. Wang, H. Tang, C. Guo, and Z. Xiu. Entropy optimization of scale-free networks' robustness to random failures. *Physica A: Statistical Mechanics and its Applications*, 363(2):591–596, May 2006.

[242] X. Wang, B. Thijssen, and H. Yu. Target essentiality and centrality characterize drug side effects. *PLoS Computational Biology*, 9(7):e1003119+, July 2013.

[243] Z. Wang, M. Gerstein, and M. Snyder. RNA-seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, Jan. 2009.

[244] D. Warde-Farley, S. L. Donaldson, O. Comes, K. Zuberi, R. Badrawi, P. Chao, M. Franz, C. Grouios, F. Kazi, C. T. T. Lopes, A. Maitland, S. Mostafavi, J. Montojo, Q. Shao, G. Wright, G. D. Bader, and Q. Morris. The GeneMANIA prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Research*, 38(Web Server issue):W214–W220, July 2010.

[245] M. N. N. Wass, G. Fuentes, C. Pons, F. Pazos, and A. Valencia. Towards the prediction of protein interaction partners using physical docking. *Molecular Systems Biology*, 7, Feb. 2011.

[246] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.

[247] R. J. Weatheritt, N. E. Davey, and T. J. Gibson. Linear motifs confer functional diversity onto splice variants. *Nucleic Acids Research*, 40(15):7123–7131, Aug. 2012.

[248] J. Weston, A. Elisseeff, D. Zhou, C. S. Leslie, and W. S. Noble. Protein ranking: From local to global structure in the protein similarity network. *Proceedings of the National Academy of Sciences*, 101(17):6559–6563, Apr. 2004.

[249] W. Willinger, D. Alderson, J. C. Doyle, and L. Li. More normal than normal: scaling distributions and complex systems. In *Proceedings of the 36th conference on Winter simulation*, pages 130–141. Winter Simulation Conference, 2004.

[250] L. Yen, F. Fouss, C. Decaestecker, P. Francq, and M. Saerens. Graph nodes clustering based on the Commute-Time kernel. In Z.-H. Zhou, H. Li, and Q. Yang, editors, *Advances in Knowledge Discovery and Data Mining*, volume 4426 of *Lecture Notes in Computer Science*, pages 1037–1045. Springer Berlin Heidelberg, 2007.

[251] N. Youngs, D. Penfold-Brown, K. Drew, D. Shasha, and R. Bonneau. Parametric bayesian priors and better choice of negative examples improve protein function prediction. *Bioinformatics*, 29(9):btt110–1198, Mar. 2013.

[252] A. Yu, C. Zhao, Y. Fan, W. Jang, A. J. Mungall, P. Deloukas, A. Olsen, N. A. Doggett, N. Ghebranious, K. W. Broman, and J. L. Weber. Comparison of human genetic and sequence-based physical maps. *Nature*, 409(6822):951–953, Feb. 2001.

[253] H. Yu, P. Braun, M. A. Yildirim, I. Lemmens, K. Venkatesan, J. Sahalie, T. Hirozane-Kishikawa, F. Gebreab, N. Li, N. Simonis, T. Hao, J.-F. F. Rual, A. Dricot, A. Vazquez, R. R. Murray, C. Simon, L. Tardivo, S. Tam, N. Svrzikapa, C. Fan, A.-S. S. de Smet, A. Motyl, M. E. Hudson, J. Park, X. Xin, M. E. Cusick, T. Moore, C. Boone, M. Snyder, F. P. Roth, A.-L. L. Barabási, J. Tavernier, D. E. Hill, and M. Vidal. High-quality binary protein interaction map of the yeast interactome network. *Science*, 322(5898):104–110, Oct. 2008.

[254] H. Yu, P. M. Kim, E. Sprecher, V. Trifonov, and M. Gerstein. The importance of bottlenecks in protein networks: Correlation with gene essentiality and expression dynamics. *PLoS Computational Biology*, 3(4):e59+, Apr. 2007.

[255] X. Yu, J. Ivanic, V. Memišević, A. Wallqvist, and J. Reifman. Categorizing biases in high-confidence high-throughput protein-protein interaction data sets. *Molecular & Cellular Proteomics*, 10(12):M111–012500, 2011.

[256] Q. C. Zhang, D. Petrey, L. Deng, L. Qiang, Y. Shi, C. A. Thu, B. Bisikirska, C. Lefebvre, D. Accili, T. Hunter, T. Maniatis, A. Califano, and B. Honig. Structure-based prediction of protein-protein interactions on a genome-wide scale. *Nature*, 490(7421):556–560, Oct. 2012.