Network Medicine Characterisation of Genetic Disorders by Propagation of Disease Phenotypic Similarities

Juan José Cáceres Silva

Supervisor: Prof. Alberto Paccanaro

Department of Computer Science
Royal Holloway, University of London

This dissertation is submitted for the degree of
Doctor of Philosophy

September 2018

Declaration

I Juan José Cáceres Silva hereby declare that this thesis and the work presented in it is entirely my own. Where I have consulted the work of others, this is always clearly stated.

Juan José Cáceres Silva
September 2018

Acknowledgements

I am very grateful to my supervisor Prof. Alberto Paccanaro, who served as an example to follow for many years. We have shared frequent meetings and discussions along the past 4 years, even during a handful of research trips abroad. Alberto has granted me his friendship, support and guidance, and I hope to have learned a thing or two from his work ethics and kindness.

I would like to thank the staff and students for making our department a nice environment. I thank Nuno Barreiro for his support and trust in many academic projects. I thank Prof. Dave Cohen for his contagious enthusiasm and interest for all forms of science and technology.

I thank my fellow permanent and visiting labmates at the PaccanaroLab, Dr. Horacio Caniza, Dr. Alfonso Romero, Mateo Torres, Diego Galeano, Dr. Cheng Ye, Rubén Jimenez, Jessica Gliozzo, Michele Nacucchi, Víctor Yubero and Santiago Noto for the work, the sleepless nights at the lab, the grind for funding, the interesting discussions and the silly moments we shared these four years. I feel obliged to give a further acknowledgement to Horacio for his help and advice, and without whom I would have not applied for this PhD. I also thank all the friends I found during these years that made me feel at home, I will dearly cherish the many memories we shared.

Last, but not least, I am grateful to many dear people back home for their unending affection and warmth. To Claudia, for being an example of courage. To my family, Ercilia, José, Tatiana, Rocío and María, for their love. To my dad Víctor, who inspired me to study and gave me all the opportunities when I needed them the most. To my aunt María Esther, who had my back so often when I was far.

Abstract

The elucidation of the genetic causes of diseases is central to understanding the mechanisms of action of a pathology and the development of treatments. Disease gene prediction methods streamline the discovery of the molecular basis for a disease by prioritizing genes for experimental validation. Technological advances, such as high throughput sequencing and screening techniques, have led to an increasing accumulation of genomic data. Despite this growth, the mechanisms of action through which genomic variants drive disease development are not fully understood. Earlier approaches to find non-experimental disease gene associations such as linkage analysis or genome-wide association studies, produce either limited results or hundreds of candidates, making experimental validation expensive and time consuming.

Modern biological networks have been exploited to capture significant features of the highly complex protein interactions, leading to the rise of computational methods in network medicine. Recent network medicine based approaches bypass the lack of functional annotation by drawing inferences from interaction data. My approach, called Cardigan (ChARting DIsease Gene AssociatioNs), is based on a semi-supervised algorithm that propagates labels on the interactome. These labels integrate disease phenotypic information expressed as a similarity measure between diseases, which is obtained by mining and comparing MeSH terms relevant for each disease on the MeSH ontology. Thorough experimentation shows that Cardigan vastly outperforms state-of-the-art disease gene prioritisation methods. This work additionally presents an exploratory extension of the approach, which, to the best of my knowledge, allows network methods for the first time to handle protein interfaces.

As a ramification of disease characterisation, this work presents an analysis of viral induced lymphoid malignancies on mice. In particular, the characterisation of the clonality of viral insertions to classify different stages of lymphomagenesis. The results show that we can identify rare driver mutations from late stage samples, with infrequent occurrences as clonal mutations, by adding statistical support of their occurrence as subclonal mutations. Several known rare cancer drivers were found to appear as subclonal mutations in late stage cancer samples more often than expected by random chance. Another research ramification focuses on a pipeline to infer

drug cocktails for the chronic phase of Chagas Disease, which were assembled from drugs with prospective efficiency against the parasite. The drug set is obtained by homology analysis of known drug targets and enzymes found in inferred metabolic pathways for the pathogen, and a random forest model trained with a large compound essay against the pathogen.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Motivation

The elucidation of the genetic causes of diseases is a central element to understand the mechanisms of action of a pathology, not only for the development of diagnosis or treatments, but also for a deeper understanding of cell biology. High throughput sequencing and screening techniques have led to an increasing accumulation of genomic data [129, 168]. Despite this growth, the mechanisms of action through which genomic variants drive disease development are not fully understood [209]. As genomic alleles and malignant mutations are continuously sequenced, most of them still miss a functional annotation. While traditional approaches fail to unveil the causes, Genome-Wide Association Studies (GWAS) often produce hundreds of candidates, making experimental validation expensive and time consuming.

Modern network medicine methods to predict disease genes improve upon previous approaches by exploiting biological networks to capture significant features of the highly complex protein interactions [213]. Generally speaking, these computational methods streamline the discovery of molecular basis for a disease by establishing a gene priority for experimental validation [3, 90, 231]. A subset of prioritised genes can be used to reduce the search space of the genome, such as in pathway analysis [77]. This reduction in search space greatly increases the sensitivity of the experimental validation, which allows genome wide enrichments of $p < 0.05$ to be significant [43], compared to enrichments of $p < 5 \times 10^{-8}$ required in GWAS [66].

Most network medicine methods relay on the *guilt-by-association* principle and prioritise to a certain degree genes which are close to disease genes. Approaches which exploit this idea differ in how they quantify the distance between candidate genes and known disease genes in the interactome [213]. Common measures for

the proximity are the number of direct connections, the length of shortest paths and diffusion kernels, including random walkers with restart and propagation flow. Furthermore, some approaches add other biological networks, or add disease phenotype information to enhance their performance.

Disease phenotype is a particularly rich and interesting data source, as well-known public databases are being constantly updated. For instance, the Online Mendelian Disease in Man (OMIM), which focuses on the association of genes to disease, comprises over 8,500 genetic disorders and 15,100 genes [125]; and ClinVar, which focuses on the association of genetic variation to phenotypes, includes records of over 440,000 unique variants and 30,000 diseases [102]. However, most methods that use disease phenotype use outdated pre-calculated matrices, which also limits the amount of diseases they can predict [206, 135, 107].

An important point to be made here is that there are many molecularly uncharacterised diseases, for which no disease gene is currently known – as of 2018 these comprise 3,359 diseases in OMIM – i.e. 39% of the entire OMIM database. Henceforth, molecularly uncharacterised diseases are called *uncharted*, while those diseases for which at least one disease gene is currently known are called *charted*.

This work presents a disease gene prediction method that predicts disease genes for both charted and uncharted diseases, and can also predict disease modules. Our approach, which we have called Cardigan (ChARting DIsease Gene AssociatioNs), is based on a semi-supervised algorithm that propagates labels on the interactome using a diffusion kernel. These labels integrate disease phenotypic information expressed as a similarity measure between diseases, which is obtained by mining and comparing MeSH terms relevant for each disease on the MeSH ontology. Furthermore, we show that Cardigan outperforms state-of-the-art methods in disease gene and disease module prediction.

This work also explores an extension which allows the usage of the Cardigan methodology to predict genes using protein interfaces and known genetic variants. A recent database of computationally predicted protein interfaces called Interactome INSIDER [130] is mined to produce a PPI where proteins are linked through interfaces. The extension has two variants: the first modifies the diffusion kernel used in Cardigan to embed the interface information into a protein-protein network, and the second proposes a diffusion method for a network with interfaces. This extension enhances the performance of Cardigan on charted diseases, showing the potential of a new data source for network medicine.

In addition, this work presents research I have done as part of two collaboration projects with different wet labs. My collaboration for the first project is part of a large analysis of a mouse lymphomagenesis study [216]. In particular, my work

consisted in the characterisation of lymphoma stages through the distribution of insertion clonality, and over-representation analysis to support the selection of novel candidates. The second project presents a pipeline to produce putative drug cocktails effective against Chagas disease in its chronic phase.

1.2 Prequels of computational medicine

For the longest period of human history, the practice of medicine was regarded with mysticism. Although our ancestors could understand that wounds and broken bones could heal, diseases were not understood until scientific progress provided the microscopes which revolutionised the field. We discovered that the basic unit of life was the cell, which composed all complex organs. The cells themselves work thanks to some small machines they produce in the form of chains of amino acids – i.e. proteins. The protein activity surprisingly proved to behave in a mechanical way; it is determined by the three dimensional structure, and the structure depends on the amino acid sequence. We could also understand that different organisms could be characterised from their chromosomes, which encode the genes that define the set of proteins available to be produced in the cells.

Although these concepts are well established, the sheer number of elements involved in the cell biology makes the study of biomolecules an arduous process. Simple organisms such as E. Coli has an estimated of 11,000 genes and 2 million proteins in the cell at any given moment [133]. While humans neither have the largest amount of genes, nor carry the largest amount of protein molecules in their cells; the human genome contains about 20,000 to 40,000 [30] genes, and 1 to 2.3 billion protein molecules per cell at any given time [133]. Furthermore, not all proteins are expressed all the time: many proteins are only produced as a response to particular stimulus, and in complex organisms different cell types also produce different subsets of proteins.

Nonetheless, early models of biomolecule interactions, activity or governing principles were built without the aid of computers. Some of them even precede genetic discoveries, such as the trait inheritance laws established by Mendel [48], where the offspring was found to acquire a trait from its parents. After extensive analysis of plant breeding, traits (later discovered as genetic alleles) were determined to be either dominant or recessive, and each individual would inherit two factors (later genes), one randomly from each parent, and that the dominant trait would be expressed. Likewise, the transformation of air and food nutrients to cell-usable metabolites led to the idea of metabolic pathways – that is, several proteins establish

a network of many reactions that lead to the production of desired metabolites [34]. Once organelles could be isolated, enzyme extracts could be obtained and individually tested on the degradation of substrates. Similar processes with exhaustive testing based on enzyme isolation and expression analysis established the initial procedures to find transcription networks, signalling pathways and disease gene associations. In particular, disease gene associations were established through linkage analysis, which establishes the likelihood of observing a phenotype and particular genetic alleles in an organism compared to random chance [149].

1.3 The advent of big data

High throughput sequencing and screening techniques changed dramatically the amount of genetic data available to study, and pushed forward our understanding of biology. Starting with the gargantuan 3 year long effort to complete the first sequenced human genome in 2001 [30, 208], and the following efforts to sequence model organisms, technology has rapidly evolved reducing the limitations of these initial efforts. The cost of a human genome sequence decreased 100-fold from the original estimated over a billion pounds by the end of the Human Genome Project in 2008 [177], with improvements to the traditional Sanger sequencing. Nowadays, the cost of a raw sequencing method is less than £2,000 [168] with high throughput sequencing (HTS) platforms.

Most modern HTS platforms follow a pipeline "template preparation, clonal amplification, followed by cyclical rounds of massively parallel sequencing. The specific strategy employed by each platform determines the quality, quantity and biases of the resulting sequence data and the platform's usefulness for particular applications."[168]. We can roughly classify the HTS platforms according to the technology, inputs and outputs. The popular Illumina analyzers amplify DNA fragments on the surface of a glass slide, and sequences based on fluorescent base incorporation, while ThermoFischer's Ion Torrent analyses the changes pH as the DNA is extended to sequence the incorporated bases. After the genome of a species is already characterised, individuals can be sequenced to analyse particular phenotypes. According to the focus, studies range from small scale gene panels or biomarkers, to Whole Exome Sequencing (WES), which target the protein coding regions (about 2% of the genome), and Whole Genome Sequencing (WGS) as the most comprehensive studies.

The availability of many sequenced individuals has helped the study of genetic diseases. The traditional linkage studies became a powerful tool for diseases depen-

dant on high penetration alleles, more so when the parents and the offspring with the phenotype could be sequenced. The traditional linkage studies were further extended to Quantitative Trait Loci mapping which can detect interactions between loci of different chromosomes. While these linkage studies largely failed to detect multifactorial and heterogeneous diseases, Genome Wide Association Studies (GWAS), which work on logistic regressions allow low penetration alleles to be discovered, however they generally require a much larger amount of samples.

Although the highly praised GWAS greatly advanced our understanding about some complex diseases such as diabetes, to which it contributed in almost 20 significant loci, they failed to deliver the characterisation of complex diseases it originally intended. Most of single nucleotide polymorphisms (SNPs) do not largely effect the overall probability of a disease, and their contribution is not detectable in GWAS. GWAS rely on the linkage disequilibrium principle [209] (the same used in linkage analysis), which can only identify loci to be strongly correlated to a phenotype [149]. Moreover, most of the genetic variation showed by the population cannot be interpreted just through GWAS, which is estimated to account for 20%, and up to 50% in the best cases, when all SNPs are considered together [209].

Network based methods came as an alternative to GWAS studies to explain diseases after modelling the cell biology.

1.4 Modelling the information

From a network medicine point of view, cellular components and their interactions are seen like networks or graphs, where the cellular entities are the nodes and the interactions edges. Commonly studied cellular networks are: *protein-protein interaction networks* (PPI), where the nodes are proteins and edges represent their physical interaction, *metabolic networks*, where nodes are metabolites and edges indicate their participation in a common biochemical reaction (also known as pathways), *regulatory networks*, where a directed edge represents a regulatory relationship between a transcription factor and a gene and *RNA networks*, where nodes can be microRNAs or interfering RNAs linked to each other (RNA-RNA edges) or to DNA (RNA-DNA edges) [9].

Manually curated databases are available, and serve to guide disease gene research. The OMIM website provides known disease and gene mappings [125]. The Gene Ontology (GO) provides an ontology of gene functions, and association files to map genes of different organisms into the ontology [7]. Probably human PPIs have the largest amount of different manually curated databases, such as the Münich

Information Center for Protein Sequence (MIPS), protein interaction database, the Biomolecular Interaction Network Database (BIND), the Database of Interacting Proteins (DIP), the Molecular Interaction database (MINT),and the protein Interaction database (IntAct), the Biological General Repository for Interaction Datasets (BioGRID) and the Human Protein Reference Database (HPRD), the Human Integrated Protein-Protein Interaction rEference (Hippie), among others [9, 175]. The Kyoto Encyclopedia of Genes and Genomes (KEGG) is the main referral for metabolic networks, but the Biochemical Genetic and Genomics (BIGG), the Human Metabolome Database (HMDB) and Reactome also are commonly used databases [9, 220, 33]. Regulatory networks are available through the Universal Protein Binding Microarray Resource for Oligonucleotide Binding Evaluation (UniPROBE) and JASPAR, but this type of networks are considered to still be the most incomplete [9]. Some microRNA-gene networks are available in databases such as TargetScan, PicTar, microRNA, miRBase and miRDB [9].

Biological networks present some interesting topological properties that reveal organizing principles, which are thought to stem from evolutionary reasons. The following are some of the main properties common to most biological networks:

- *Scale-free distribution*: the edge degree distribution of PPIs and metabolic networks follow a power-law, defined as $P(k)\ k^{-\gamma}$, where k is the edge degree and γ is the degree exponent of the network. A consequence of this distribution is the existence of hubs (highly connected nodes) that dominate the structure of the network. This is known as the Barabasi-Albert model, which became the standard PPI model, over the random Erdös-Renyi model [5].

- *Small-world*: the longest path between any two nodes in the networks is relatively short (i.e. a protein is only a few interactions of any other protein in a PPI) [215]. This effect can be seen as a consequence of the degree distribution of PPIs. The main visible consequence, in a biological sense, is that the perturbation in any node can affect its entire neighbourhood, which in turn may affect the behaviour of the overall network [10].

- *Modules and network bridges*: a high degree of modularity is present in most of the networks; areas around hubs are likely to be highly connected. Although most modules are highly overlapping, some low degree nodes connect big modular regions – i.e. the network bridges [62, 197].

- *Motifs*: particular small network subgraphs are observed to be under or over represented in biological networks. These subgraphs are likely to be associated

highly specialized functions, such as negative feedback loops or biological oscillators [10].

- *Guilt-by-association*: interacting genes are likely to perform a similar function. These interactions usually imply that the interactive proteins participate in common cellular functions, as they are part of common pathways, or form compounds [145].

A great amount of computational methods in network medicine derive from the application of the *guilt-by-association* principle taking into account the topological properties of the biological networks. Some notable applications are described in Section 2.4.

An approach to use the network motifs, is to transform the PPI network into a network of small subgraphs (graphlets), and analyse their distribution and connections. Geometric random graphs [153] were shown to be good models for the graphlet distribution in eukaryotic PPI networks [163], and graphlet networks can be used to discover motifs associated with biological processes, and network functions [132].

However, most computational methods utilise machine learning to exploit the network structure in different applications, such as gene function [136, 184], protein complexes [2, 140], or disease genes [148, 206, 186]. Relevant machine learning methods and their relation to disease gene prediction are presented in Chapter 2.

1.5 Contextualising the network medicine paradigm

A thorough understanding of the genetic characteristics and their implications for human traits and conditions is far from simple. Millions of variations occur naturally in the human genome and are responsible for many phenotypes, such as physical traits, drug response or disease [142]. Despite many of these notions are generally understood, a precise definition may be impossible to accommodate all the required biological facts. Furthermore, these concepts need to be set in the context of network medicine to follow the paradigm of this work.

The human genome is a complex structure that presents a noticeable heterogeneity when comparing individuals. The most common variation consists in the mutation a single nucleotide, roughly every 300 nucleotides – i.e. humans are expected to have 10 million single nucleotide polymorphisms (SNPs) [69]. While SNPs occur in coding and non-coding regions of the genome, observable phenotypes are generally associated only to SNPs within genes or regulatory regions of the genome [69].

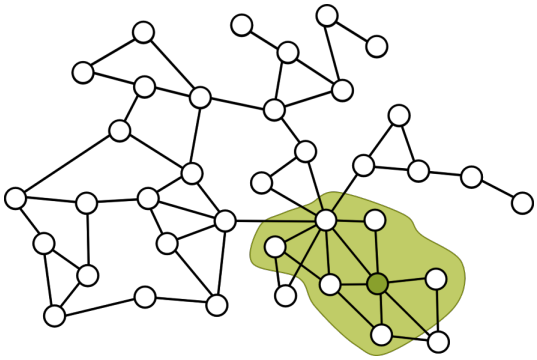


Fig. 1.1 **A disease module in an interactome.** The nodes in the network represent genes (or proteins), and the edges represent interactions between these genes. The most common type of interactomes used in the area are protein-protein interaction networks, which are connected by edges if they physically interact. A known disease gene is coloured and the perturbed area in the network affected by the variant is highlighted – the disease module.

Variant forms of a gene that appear with a significant frequency on the population (i.e. 1%) are known as alleles, and may include multiple SNPs. It is important to notice that some alleles are expected to produce the same trait, and only a fraction of the known traits represent deleterious mutations [102].

Although a medically correct definition for disease is a complicated affair, the general notion will suffice for the understanding of this work. Without technicality, a disease is a disorder with specific characteristics which produces a negative impact in an organism. As the focus of this work is on genetic diseases, the term disease will not include illnesses produced by external factors, such as bacteria or viruses, unless specifically stated. Note that with this notion, the term disease is used as a generic label which also includes genetic syndromes and phenotypes.

Simple genetic diseases are known to be caused by a single gene with a mutation. Complex diseases include multiple mutations and potentially different combinations of these mutations [63], and the onset of some complex diseases are known to require one of several combinations of low penetration mutations [63]. Furthermore, environmental factors may also play a role for the onset and development of a disease [37]. Taking into account all these confounding issues, in many cases it is still impossible to determine the exact combination of factors that cause complex diseases. This work uses a common denomination, in which a disease gene is the cause of a single gene disorder or a high penetration mutation for a complex disease.

Regardless of their number, disease genes rarely act in isolation. A disease module can be described as all genes and proteins affected during the disease process. From a network medicine paradigm, where the expected functions in an

organism are represented by standard biological networks, diseases are seen as perturbations in a network. The areas affected by the perturbations are known as the disease modules. However, the literature is far from a universal consensus about the definition of disease module (or what means for a gene to be involved with a disease). For instance, publications from the well known Barabasi group uses the network medicine interpretation [10, 9], or may refer to the known molecular basis as the disease module [127], or resort to the accumulation of genes known to be associated to similar phenotypes [55] to capture a more comprehensive module. The molecular basis interpretation appears to be aligned with a more frequent approach for wet-lab publications, which build modules by collecting differentially expressed genes [39, 44, 166]. Additionally, publications about complex diseases might consider the module as the collection driver and statistically significant passenger mutations [63, 51]. This work allows the concept of disease module to the wider comprehension of the term.

Following the network medicine paradigm, we can see that disease modules describe the disease genotype, and correlate with topological modules in the network [55]. Therefore, computational methods quantify distances within a machine learning approach to prioritise candidate disease genes according to their closeness to known disease genes. For example, Oti *et al.* [148] use direct neighbours, Köhler *et al.* [90] use random walkers with restart, and Navlakha *et al.* [139] include propagation flow and clustering techniques. Other types of data are also informative of the location of disease genes such as phenotype, gene expression, gene ontology, metabolic pathways, and for this reason have been included in different methods. One group of methods integrates the data into a unique graph that is then used for the prediction. Lage *et al.* [99] include disease phenotype in the form of clinical features extracted by text mining from scientific papers; Wu *et al.* [222] create binary networks where nodes represent genes, and these are connected when their BLAST [6] E-value is higher than a predefined threshold; Chen *et al.* [25] include information from the Gene Ontology, the Mammalian Phenotype [188] and various types of pathway annotations; Li *et al.* [107], Vanunu *et al.* [206] and Mordelet *et al.* [135] include the van Driel disease similarity information [205] to enhance the network; and other authors use heterogeneous networks where nodes can be either diseases or genes – Xie *et al.* [223] connect the nodes with OMIM and MGI mouse phenotype- gene associations [45], and Zeng *et al.* [231] use HeteSim scores [185]. Another group of methods carries out inferences for each different type of data separately, and then integrate the results. In particular, Aerts *et al.* [3] use co- expression networks, metabolic pathways, Gene Ontology, among others; Franke *et al.* [50] include the Gene Ontology and co- expression networks; Radivojac *et al.* [165] use the Gene

Ontology, the Disease Ontology [179], and features based on protein sequence; Karni *et al.* [83] use disease based co-expression networks; and George *et al.* [54] use metabolic pathways and Pfams [13].

1.6 Contributions

- It presents a state-of-the-art network-based disease gene prediction method, named Cardigan, which is shown to outperform various state-of-the-art methods. Cardigan allows the prediction of genes for both molecularly uncharacterised and molecularly characterised diseases, through the addition of plausible genes obtained by disease phenotype analysis. Notably, tying the prediction to the inclusion of a phenotype similarity does not reduce the coverage of the predictable diseases from the OMIM database.

- It presents a novel exploratory analysis about the usage of graphs of protein interfaces for the prediction of disease genes. The approach presents a new diffusion method, and a procedure to assemble the protein interface graph from a state-of-the-art database of inferred protein interactions at residue level. This exploratory analysis is, to the best of my knowledge, the first network medicine application to use a protein interface graph. The usage a protein interface graph appears to be a promising source of information for disease gene prediction methods.

- It provides a further insight in the complex relation between disease modules and the *guilt-by-association* principle. Disease phenotype similarity appears to capture additional features than protein function which allows the prediction of common genotype. Additionally, it shows that while drug targets are diverse, etiological drugs form modules which can help in the characterisation of disease modules.

- It presents a procedure to assemble a heterogeneous interactome that combines protein-protein interaction networks from different organisms, which shows utility for the analysis of experiments on model organisms in the context of human genetics. This proposal also shows applications as a simple edge completion technique.

- It presents several computational techniques used in collaborative projects which contain wet lab components. The first project is analysis of viral induced lymphoid malignancies on mice, which yields candidate genes for the disease.

Here, it describes the usage of entropy of mutation clonality for the classification of tumour stage. The second project presents a computational pipeline to predict drug cocktails for Chagas disease. This shows the production of a putative drug set by homology-based and machine learning approaches, and the assembly of drug cocktails by a multi-objective approach.

1.7 Structure of the book

Chapter 2 presents an overview of the machine learning approach to computational biology. In particular, it develops deeper into the semi supervised learning paradigm on graphs and its application to modern biological problems, such as protein function prediction, phenotype characterisation and disease gene prediction.

Chapter 3 presents the state-of-the-art disease gene prediction methods, and discusses current strengths and weaknesses of the competing approaches.

Chapter 4 describes the experiments, evaluation procedures and datasets used for the validation of the different methods presented in this work.

The following three Chapters (5, 6 and 7) are the core contributions of this work, and present a sequence of intrinsically connected methods for the characterisation of disease. Ordinal (Chapter 5) describes a method to identify disease modules based exclusively on disease phenotype data, and sets the basis for a prediction method which includes a network component to the procedure. Then, Cardigan (Chapter 6) presents a semi supervised disease gene prediction method, which incorporates phenotypic information to predict uncharted diseases on standard interactomes. Finally, an extension of Cardigan (Chapter 7) is presented as an exploratory analysis to improve Cardigan's performance by exploiting a network with interfaces, built from a state-of-the-art database of inferred protein interface interactions. This Chapter also presents a theoretical framework for label diffusion on networks with interfaces.

Chapter 8 presents a compilation of additional research I have done related to disease characterisation. The Lymphomagenesis and Chagas projects are collaborations with other research groups. The analysis on the Human-Mouse interactome and the drug module analysis stem from these projects. LanDis is a web based application used to visualise and explore a disease landscape established by phenotype similarity.

Chapter 9 discusses the overarching contributions of this work on the field of computational biology and machine learning. Notice that Chapters 3, 5, 6 and 7 include discussions about their particular topics to preserve the context.

Chapter 10 presents an application in development derived from Cardigan, and interesting additional projects that appear to be viable follow ups of Cardigan and the usage of protein interface networks.

Appendix A is a manual to use the Cardigan software. Cardigan is given as a library for Python 2.7, and the manual includes code snippets to produce the experiments used in this work. Appendix B provides categorical and identifier mappings produced for particular applications presented in this work.

Chapter 2

Learning with biological networks

This Chapter intends to contextualise the network medicine approach of disease gene prediction within machine learning, and present some of the main learning techniques used in state-of-the-art disease gene prediction methods.

First, Section 2.1 presents an overview of Machine Learning methods and presents some benefits of graph methods (i.e. network medicine methods) when compared to other approaches. Then, Section 2.2 explores the production of PPI networks, which in turn becomes the main biological network used in this work. Thereupon, Section 2.2 introduces the rich history of graph theory and adds support to the usage of graphs as prime data structures for machine learning. It also presents the intuition and the essential formulation of two widespread approaches: *graph diffusion* and *Support Vector Machines* (SVMs).

Additionally, Section 2.4 shows that network medicine methods are relevant for several biological applications, such as the identification of protein complexes [140], drivers for drug response [40], network architectures [65, 119], disease genes [213], and genome annotation [228, 79].

2.1 Fundamentals of learning

A holistic description of a machine learning can be thought as any type of algorithm related to artificial intelligence: cognitive simulation, expert systems, model generation, knowledge representation, among others [131]. However, as this work focuses on data analysis, only methods focused on predictions of "hidden" information are considered. These methods can be thought of as mathematical models to unveil regularities from the data [14]. Common applications are data classification, dimensionality reduction, regression and modelling [182].

The mathematical representation of the available data is an input vector, and the outcome of the method is an output vector [182]. Usually, these methods require a separation of the data to avoid biases in the predictions. Data used for the construction of the model is known as *training*, while the data used for the validation is known as *test*.

A helpful way to analyse machine learning approaches is to divide them according to the training data available. This division is not trivial, as the type of data available also determines the possible applications for the learning procedure. The main difference stands from the availability of the expected outputs for particular inputs, known as *targets*:

- **Supervised Learning**: input vectors and target vectors are available in the training and testing sets. This situation can be understood as the search for a translation function between these sets. Classification and regression are the traditional applications for supervised learning: when the input vector is being translated to a discrete category, the problem is known as *classification*; when the output is translated to some continuous range, the problem is known as *regression* [14].

- **Unsupervised Learning**: the training and testing sets contain no known targets for any input. As less information is available, under these circumstances the applications are traditionally designed to uncover the structure of the data. Common applications are clustering, visualisation and modelling: *clustering* applications characterise the data by segregating inputs into groups of similar elements [14]; *dimensionality reduction* problems produce projections of high-dimensionality data onto low-dimensionality spaces, traditionally for data visualisation or as a pre-process for further learning [14]; *modelling* problems search for a generative method to produce elements which fit the distribution of the input set, for instance as a density estimation [182].

- **Semi-supervised Learning**: some of the input vectors have target vectors associated, while most of the input vectors have none. Applications for semi-supervised learning methods are shared with supervised or unsupervised learning through extensions to account for the diversity in the data. Common applications are semi-supervised clustering and semi-supervised classification. *Semi-supervised clustering* is basically a boost for unsupervised clustering algorithms which includes known examples of some clusters. While most clustering problems do not favour a particular type of learning, anomaly detection has a prominent semi-supervised clustering approach [22]. *Semi-supervised*

classification fundamentally changes the supervised approach, as missing targets must be accounted for. For this reason this application derived into several subdivisions [237]: *generative models* when the input is generated by an identifiable mixture model, *self training* when the classifier uses its own predictions to train itself, *co-training and multiview learning* when multiple classifiers are used and they cross train other classifiers, *avoiding changes in dense regions* when the input data can be kept in a space where regions of different density can be appreciated.

The production of datasets inherently comes with a cost, while in many contexts the acquisition of unlabelled data is cheap, inputs with known targets are generally scarce [182]. The production of targets can be particularly difficult for computational medicine datasets, such as disease development, rare trait expression, drug response or survival rates. For this reason, semi-supervised methods are of particular interest in this area.

An orthogonal classification for learning discerns when method performs an inference by building a model, known as *induction*, or when the inference is performed in a case by case basis for the unlabelled data, or *transduction* [53]. While the learning methods can be inductive regardless of the availability of known targets, the transductive approach is inherent to semi-supervised learning; in particular, many *graph based methods* have a natural transductive inference [23]. Note that graph based methods are not exclusive to a particular type of learning procedure; the name only implies the usage of data which holds a graph structure. However, if the problem domain holds the *guilt-by-association* principle, a manipulation of local and global properties of the manifold can produce effective propagations of inferences within the unlabelled data [235].

This work considers the task of predicting disease genes as a semi-supervised problem using a graph based method. The prediction is given as a ranking of all genes within a PPI network, where a missing gene is meant to be ranked as high as possible.

2.2 Assembly of interactomes

Modern PPI networks usually come from the curation of high-throughput techniques, such as yeast two-hybrid (Y2H) and co-affinity purification followed by mass spectrometry (AP-MS). These experiments produce a confidence value for the interaction between pair of proteins, which usually needs a further processing to obtain the high quality interactions [202, 95, 192, 171, 161, 28]. As the results of these experiments

are intrinsically noisy, networks rendered by simple thresholding are too sparse or of low quality. Studies show that Y2H screens yield false negative rates of 43-71% and false positive rates of 64% [42]. AP-MS screens somewhat improve on the false negative rates, which range between 15-50%, however, false positive rates range between 63-77% [67].

In order to decrease these error rates, computational techniques were used to clean the resulting data based on expected properties of PPIs [115, 64, 28]. Additional techniques developed generative models to improve the classification errors (i.e. reduce the false positive and false negative ratios) of interactions based on experimental data [229, 96]. For instance, some improved the association errors by using Bayesian classifiers [115, 28], Gene Ontology annotations [118], topological features [229], and random graph models [162, 96].

High quality PPIs are frequently aggregated into protein-protein collections for different organisms in publicly available databases [160, 24, 175, 195, 178]. These databases are provided by different groups with their own curation standards; they provide different information under many formats [233]. For instance, they may include binary or weighted edges, overlapping data, conflicting identifiers, and various evidence types. In weighted networks the links between two proteins are labelled with a weight whose value is related to the probability of the interaction. In binary networks, links are not labelled and a link is either present or missing (denoting the existence or the lack of interaction). Moreover, interaction data can be experimental or predicted.

2.3 Graphs and learning

The origins of graph theory take place hundreds of years before the development of machine learning. This provides a rich theoretical background and a wide set of mathematical properties which become exploitable within the machine learning paradigm.

Early problems of graph theory can be dated to Euclidean geometry and are intrinsically related to combinatorial problems. It was not until Euler's analysis of the Königsberg bridges that the geometric component of the problem took a second place in favour of the node degree analysis [218]. While graphs became more complex in the representation of topological properties of higher dimensional geometry, chemical bonds and map colouring, the analysis of Hamiltonian paths can be seen to settle the link between topology and graph theory [60]. Graphs were also

linked to algebraic theory initially by Kirchhoff [218], extending its use to many areas in mathematics.

The concept of graphs is rather simple to understand, as they are composed by a set of nodes and the connections between those nodes. However, they are useful in many contexts due to their flexibility to encode information within a meaningful structure. Common graphs examples are road maps which in different scales connect houses, cities and countries; atoms connected by bonds; houses connected in tree like structures to local electricity distribution systems, then to city, and even nation wide providers; cellular components connected by different interactions; people connected by social relations; web pages connected by hyperlinks; among others. Graphs can also be designed to convey knowledge in the form of ontologies, for instance the taxonomical classification of species, categorisation of medical information, language semantics, and others. Graph data can be rich and complex, and there is no universal method of analysis. Nodes and edges may be heterogeneous, and the expected output of the process is tied to the nature of the data at hand. This derived in the development of machine learning approaches suitable for different applications.

Among the many approaches, Graph Diffusion and Support Vector Machines are presented as they are the core for several reference disease gene prediction methods. Interestingly, both approaches derive in the usage of Kernels [14] by different constructions.

2.3.1 Graph diffusion

The main idea behind graph diffusion is that some nodes contain a value which must be passed to its neighbours, and is an analogy of the a physical diffusion process. The original formulation of a diffusion process comes from Fick's laws [47]. The first law states that the flux J is proportional to the negative of the concentration gradient:

$$J = -D\nabla c,$$

where D is a diffusion coefficient, and c is the concentration. The second law states that the rate of change of concentration is proportional to the second derivative of the concentration:

$$\frac{\partial c}{\partial t} = D\nabla^2 c, \tag{2.1}$$

where t is the time variable and ∇^2 is the Laplace operator or Laplacian. Equation 2.1 has the same formulation as the well known Heat equation by swapping the

concentration c by heat f, and the diffusion coefficient D for thermal diffusivity α:

$$\frac{\partial f}{\partial t} = \alpha \nabla^2 f. \tag{2.2}$$

Then, setting the heat as a function of the position $f(x)$ within an m dimensional object, gives the extended notation for the Laplacian:

$$\nabla^2 f(x) = \frac{\partial^2 f(x)}{\partial^2 x_1} + \cdots + \frac{\partial^2 f(x)}{\partial^2 x_m}. \tag{2.3}$$

The following transformation is made to account for the discrete nature of the graphs. Taking $m = 1$ yields a 1-dimensional object, where the finite element approximation of the Laplacian is:

$$\nabla^2 f(x) = f''(x) \approx \frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{(\Delta x)^2}. \tag{2.4}$$

Considering a uniform basis $\Delta x = h$, points $f(x_i) = f_i$ are $f_1, \ldots f_n$, with $f(x_i + h) = f_{i+1}$. Then, the Laplacian over the entire object is:

$$\nabla^2 = -\sum_{i=1}^{n} \frac{2f_i - f_{i-1} - f_{i+1}}{h^2}. \tag{2.5}$$

Finally, this sum can be understood as the addition of vectors $[-1 \;\; 2 \;\; -1]$ centred on i, this is:

$$\nabla^2_{<i>} = -\frac{1}{h^2} \sum_{a \in \Omega - \partial\Omega} \begin{bmatrix} -1 & 2 & -1 \end{bmatrix}, \tag{2.6}$$

where $\nabla^2_{<i>}$ is a sub-vector of 3 components centred in i, and $\Omega - \partial\Omega$ represents the interior of the object. The coefficients from the finite elements Laplacian inside the sum can be rearranged in a matrix L_{FE} as:

$$L_{FE} = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & -1 \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix}. \tag{2.7}$$

The analogy follows by considering the adjacent finite elements as the nodes in a graph. The ensuing formulation is given for an unweighted graph for the sake of simplicity, however the results are valid if the adjacency matrix A is changed for a

weighted matrix W. Given an unweighted graph $G = \langle V, E \rangle$ the graph Laplacian L (a matrix size $n \times n$), is produced by the addition of matrices:

$$L_{[ij]} = \sum_{(i,j) \in E} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \tag{2.8}$$

where $L_{[ij]}$ represents the 2×2 sub-matrix of rows and columns i and j. This definition is equivalent to the traditional formulation of the graph Laplacian given as the construction $L = D - A$, where the degree matrix D is diagonal of size $n \times n$, with non-zero elements:

$$D_{ii} = \sum_{j=1}^{n} A_{ij}, \tag{2.9}$$

and A is the adjacency matrix of graph G of size $n \times n$.

The graph Laplacian can be used as a quadratic operator $y^T Ly$ to produce:

$$y^T Ly = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}(y_i - y_j)^2, \tag{2.10}$$

where y is a vector of n components. An intuition behind this solution is given by comparing Equations 2.8 and 2.10, where the sums correspond to the same elements. Notice that $A_{ij} = 0$ if there is no edge between nodes i and j. While not identical, Equations 2.5 and 2.10 represent the flux density of the diffusion process.

The graph Laplacian can be also given in a normalised form $L^{sym} = D^{-1/2} L D^{-1/2}$, where the elements are:

$$L_{ij}^{sym} = \begin{cases} 1 & \text{if } i = j \\ \dfrac{-A_{ij}}{\sqrt{D_{ii} D_{jj}}} & \text{otherwise} \end{cases}, \tag{2.11}$$

and is used as the following quadratic operator:

$$y^y L^{sym} y = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} \left(\frac{y_i}{\sqrt{D_{ii}}} - \frac{y_j}{\sqrt{D_{jj}}} \right)^2. \tag{2.12}$$

These quadratic forms (Equations 2.10 and 2.12) can used as cost functions which have a global optimum, and serve to produce label diffusions in graphs since the matrices can be inverted [27].

Naturally, the graph Laplacian is not used for all diffusion methods. However, it suffices to consider the quadratic forms as Kernels to see similarities between multiple methods. The usage of Positive Definite kernels is widespread [136, 184, 206] as

they produce invertible matrices. Other common graph propagation approaches are exponential Kernels, tensor products, Gaussian Processes and random walks [92].

2.3.2 Support Vector Machines

The basic support vector machine (SVM) is a supervised learning method to classify positive and negative elements. The main idea is to produce a hyperplane that divides the hyperspace into the positive and the negative subspaces, and the ideal hyperplane is the one that leaves the widest gap between the known positive and negative examples – i.e. SVMs are maximum margin classifiers [14]. Formally, an n dimensional hyperplane H is characterised by a real vector w of n components and a real constant d, and is defined by all real points x in an n dimensional space such that:

$$w \cdot x + d = 0.$$

The geometrical interpretation is that all lines in H are orthogonal to w.

Ideally, this hyperplane H will separate the positive samples x^+ from the negative samples x^- in different subspaces – i.e. $x^+ \cdot w + d > 0$ and $x^- \cdot w + d < 0$. The hyperplane H that separates positive and negatives samples with the largest margin is calculated by solving:

$$\begin{array}{ll} \text{minimise} & \|w\| \\ s.t. & y_i(w \cdot x_i + d) \geq 1 \end{array},$$

where y_i is 1 if x_i is a positive label, and -1 otherwise. This formulation can be solved by calculating the Lagrangian and equating to zero [191]. The Lagrangian is:

$$\mathscr{L} = \|w\| - \sum_{i=1}^{n} \lambda_i (y_i(w \cdot x_i + d) - 1),$$

where n is the number of samples, and λ_i are the Lagrange multipliers. The derivative is zero when:

$$w = \sum_{i=1}^{n} \lambda_i y_i x_i \quad \text{and} \quad \sum_{i=1}^{n} \lambda_i y_i = 0.$$

This gives the dual formulation of the SVM:

$$\begin{array}{ll} \text{minimise} & \sum_{i=1}^{n} \lambda_i - \frac{1}{4} \sum_{i,j=1}^{n} \lambda_i \lambda_j y_i y_j (x_i \cdot x_j) \\ \text{s.t.} & \lambda_i \geq 0 \text{ and } \sum_{i=1}^{n} \lambda_i y_i = 0 \end{array}.$$

While many extensions allow SVMs to handle multiclass classification, soft separability constraints and the ability to work with unlabelled data [23], the most notable contribution is the kernel trick [68]. The kernel trick produces a high dimensional embedding of points x_i, which can be separable when in the original space they are not. The kernel formulation is trivial on the dual problem, in that it replaces the dot product $(x_i \cdot x_j)$ with a kernel $K(x_i, x_j)$.

2.4 Biological Applications

Network medicine approaches have been developed for many biological applications which shows their versatility and power. The following list of applications is not exhaustive, but serves to portray diverse scenarios where network medicine are applied.

Notice that the list does not include Disease Gene Prediction, the main application discussed in this work, as it is presented at length on Chapter 3.

Protein Function Prediction

The need of protein function prediction methods comes from the exponentially increasing gap between the amount of sequenced proteins, and the amount of annotated proteins [168]. In practice, computational methods produce a putative set of Gene Ontology terms which is used to define the protein function. Among the three Gene Ontology categories, *Molecular Function* and *Biological Process* are regarded to be of particular interest [164].

These annotations would ideally come in the most specific terms of the ontology to provide the maximum amount of information about a protein. The hierarchical nature of the ontology allows the propagation of a annotation to all its ancestors through the *is_a* or *part_of* associations.

Following the results of the critical assessment of functional annotation (CAFA) competitions [79, 164], the usage of simple sequence similarity or shared domains has proven insufficient to transfer functional annotations. The integration of multiple alignments through machine learning techniques such as SVMs proved to increase the prediction accuracy. The CAFA results show that no method is superior in all function prediction categories, and similar types of prediction methods lead to somewhat similar results (roughly speaking sequence based methods discover a cluster of functions, while PPI based methods discover another cluster). This suggests that ensemble approaches, which integrate high quality predictions, may benefit by including predictions of different method types [79].

Discovery of network architectures

In addition to protein function, the identification of protein-protein interactions and pathways are essential to characterise an organism. The prediction of these network architectures allows the study of the growing amount of newly sequenced organisms, and serves to guide experimental trials for interaction discovery [123].

It is easy to understand the benefit of the identification of missing protein protein interactions if only for the sake of network based methods, as it forms the basis of the multiple methods with biological applications. This problem can also be seen as graph edge completion, and has seen benefits from integration of annotations across different species [28, 195], Bayesian classifiers [115], GO annotations [118] and random graph models [96].

Additional techniques are used to infer dynamical biological networks, such as transcriptomic networks, or signalling and metabolic pathways. An edge between a pair of interactors can be inferred from an homologous pair of interacting proteins (an *interolog*) observed in another species [123], nodes for transcriptomic networks are identified from the genome by hidden Markov models [41] and quantification of RNA-Seq (Cufflinks algorithm) [200], and ensemble of expert predictions on different benchmark sets [119]. Similar approaches are used for pathway prediction, which are based on exhaustive extraction of biological features and ensemble prediction [35].

Protein Complex Prediction

The protein complexes are seen as dense connected components in a PPI network. However, PPI networks are incomplete [178], and sets of proteins from dense topological modules in these networks frequently only share a fraction of their functions [140]. This can be expected as proteins from the same topological module can part of more than one complex, or be involved in different biological processes [140]. Therefore, the identification of protein complexes can be seen as a process to cluster possibly overlapping groups of proteins with significant connectivity and with coherent function.

Network method approaches for the identification of protein complexes differ in the process used to collect sets of proteins, the measure to quantify the connectivity of the putative complex, and the usage of data other than PPI for further validation [190].

Based on the analysis by Srihari *et al.* [190], protein complex prediction methods can be classified in two main groups: those solely based on PPI topology, and those which use additional biological data. Some network clustering approaches

are: Markov clustering (MCL) based on random walkers [204], Clustering based on Maximal Cliques (CMC) based on iterative merges up to reliability threshold [112], Clustering with Overlapping Neighbourhood Expansion (ClusterONE) based on greedy neighbourhood expansion and a customised cohesiveness measure which accounts for missing PPIs [140], and Hierarchical Agglomerative Clustering with Overlaps (HACO) based on non unique merges of hierarchical clusters to allow complexes to overlap [211]. Some methods that use additional data are: COre AttaCHment (COACH) based on producing a small and topologically compact set of proteins that provide core function for the complexes, which are rounded up with proteins that interact with at least half of the core; and Restricted Neighbourhood Search Clustering (RNSC) which trades proteins within putative complexes until a cost function is minimised, and filters the resulting complexes based on functional coherence [89].

Characterisation of drug response

The characterisation of drug response is a rich topic and has many further applications in drug design and repositioning. The identification of drug targets can be seen as the starting point to discover a mechanism of action, which often involves metabolic pathways, signalling and transcriptomic networks [174, 176].

Computational inference of drug targets is an alternative approach to large scale component screening, aimed to lower the cost of drug discovery. Studies estimate that Pharmaceutical companies spent £ 2,000 million in 2014 for screening-based discovery, which shows an upward trend from the £ 620 million estimated in 2003 [137]. These costs can be lowered further by drug repositioning, which comes with the added benefit of identifying drugs that are approved and have cleared clinical trials.

Some approaches are devoted to produce more adequate networks, such as the integration of drug-gene interactions in various biological networks [105, 74, 227], or creation of drug-disease networks [75]. Several methods predict drug response by annotating drug features such as chemical properties, structure, molecular targets, derived gene expression and variants in the effected genotype, with neural networks and random forests [128], logistic regression [59] or integrating them to produce SVM kernels [138, 214]. Other computational methods select candidate genes for drug repositioning by clustering heterogeneous drug-disease networks [221], using Bayesian regression on drug off-target effects [80], or by minimising the loss function in a drug-disease network assembled from several drug similarity matrices and disease similarity matrices [234]. While each method presents particular advantages

and limitations, integrative approaches [59, 214] seem to improve the performance in the prediction of drug-disease associations [106].

Quantification of disease similarity

Throughout human history, medicine has focused on the diagnosis, treatment and prevention of illness and disease. Modern computerised medical support systems exploit disease phenotype information and aid the efforts of medical doctors through personalised patient recommendations, automated differential diagnosis, and other applications [86].

The characterisation of disease phenotypes involves a massive amount heterogeneous information collected with little standardisation. However, there are some reference resources that curate this knowledge, for instance some ontologies are used as *de facto* annotation standards for medical vocabulary (MeSH [143]) or gene functions (GO [7]), and others provide catalogues of human genetic disorders (e.g. OMIM [125] and UNIPROT [31]).

Multiple computational methods for diverse biological applications use disease data in the form of a disease similarity, which becomes an application on its own. Disease similarity methods score disease relatedness based on phenotype or molecular properties. Phenotype approaches mine MeSH terms and annotate diseases on the MeSH ontology and quantify the similarity based on multiple ontology measures [20, 91, 205]. Molecular approaches derive the similarity from likelihood of co-localisation of disease genes [151], and the function similarity of the disease genes [122].

Further sections reference two main phenotypic disease similarity measures: one by van Driel *et al.* [205] used in methods by other authors, and one by Caniza *et al.* [20] used in my methods. van Driel *et al.* initially model a disease with a vector of MeSH term frequencies, which contains the count of terms from the disease descriptions found in OMIM. The count vector is then transformed in a *refinement* step so that close terms in the MeSH ontology can share their counts (e.g. a child term will contribute to the counts of its parents). Each term is further scaled by the inverse frequency of each term in the disease description corpus (this enhances the contribution of the infrequent terms). The similarity between diseases is computed by the cosine similarity between their refined vectors. The Caniza similarity is based on the the Resnik similarity [167] between the sets of MeSH terms found in publications associated to each disease in OMIM (more details on Section 4.3.3). The Caniza similarity was chosen for this work as it showed to correlate the closeness between disease genotypes and phenotypes better than several

well established semantic similarities. The comparison [20] evaluated proposals by Jiang and Conrath [78], Lin [110], Pesquita [154], and van Driel [205]. Notice also that the van Driel similarity is a fixed matrix with 5,080 diseases (from the 2005 OMIM release), while the Caniza similarity provides an algorithm to calculate the similarity between all diseases found in any OMIM release.

Chapter 3

Literature review on disease gene prediction methods

Network medicine methods have been the state-of-the-art approaches for the computational prediction of disease genes. A key component of these methods is the usage of the *guilt-by-association* principle to find prospective genes: the new candidates are located close to known genes in an interactome.

This Chapter presents a review of the reference disease gene prediction methods and discusses the notable features and possible drawbacks of the approaches.

3.1 Endeavour

Endeavour is an ensemble method which combines independent inductive models given for multiple data sources into a single aggregated prediction. Each data source provides its own information (i.e. the known genes depend on the source), which is used to produce each prioritisation model. Then, the models are used to produce gene rankings, which are finally integrated by using Order Statistics [199]. Notice that Endeavour is not a network based method, however, it includes a data source integration approach, which makes it interesting to study.

Endeavour predicts genes by training the initial models with a set of genes of interest, and then evaluates other sets of genes according to how well they match the trained models. There is one model produced per data source, which requires independent training. ENDEAVOUR considers two big types of data sources: attribute-based, vector-based, and then has a some data sources with particular treatment.

Attribute-based: each data source produces a set of labels for any given set of genes. The training consists in the selection of overrepresented labels $L_1, \ldots, L_k$ for the training gene set according to Fisher's Omnibus Meta analysis. Gene sets are

evaluated independently for each label L_i and is ranked according to the resulting Fisher's Omnibus Meta analysis. The used sources are the Gene Ontology [7], EST expression [187], InterPro [230] and KEGG [82].

Vector-based: each data source produces a vector for any given set of genes. The training consists in producing the vector V for the training gene set. The evaluation consists in the Pearson correlation between vector V and the vector for the gene set. The vector-based sources are literature abstracts mined for GO terms using IDF profiles [172], microarray expression data or TRANSFAC weight matrices for cis-regulatory motif predictions [124].

BLAST [6] evaluates a test set by the lowest e-value produced when compared to the train set. BIND [8] evaluates the Jaccard coefficient between the neighbourhood 1 of the test and the train sets. And finally *cis*-regulatory module (CRM) databases are evaluated by matching motifs of the test set with transcription factor binding sites of the training set [4].

Ranks from multiple data sources are merged by using order statistics. The algorithm evaluates a set of N rankings per gene using the Q statistic. The Q statistic is calculated from a joint cumulative distribution of an N-dimensional order statistic [194]. This is $Q(r_1,\ldots,r_N) = N!V_N$, where V_N is calculated as:

$$V_k = \sum_{i=1}^{k} (-1)^{i-1} \frac{V_{k-i}}{i!} r^i N - k + 1,$$

where $V_0 = 1$, and r_i is the rank ratio of data source i, calculated as the position of gene in the ranking i divided by the number of genes in the ranking i. The Q statistics are sorted, from lowest to highest, to produce the consensus ranking. Furthermore, a one-tailed p-value corresponding to the Q statistic is added to each gene, which corresponds to the confidence of the consensus ranking [3].

3.2 PRINCE

PRINCE (PRIoritizatioN and Complex Elucidation) is a graph method which uses the *guilt-by-association* principle, and establishes a strength of association between proteins through network properties [206]. PRINCE creates a prior information vector $Y(v)$ based on the van Driel disease similarity [205]. The graph represents a human PPI network, which serves to connect genes, and allows weighted edges [58]. They propose an iterative formulation to produce a smooth score distribution over the graph. A node v receives score $F(v)$ from its neighbours $N(v)$ and its initial

value $Y(v)$ by the formula:

$$F(v) = \alpha \left[\sum_{u \in N(v)} F(u) w(v,u) \right] + (1-\alpha) Y(v)$$

where, w is $w = L^{sym} - I$ and L^{sym} is the symmetric normalised Laplacian of the graph, and $Y(v)$ is the initial score of node v, defined as $Y(v) = disease\_similarity(q,d)$ if v is associated to disease d, and $Y(v) = 0$ if no disease is associated with v [116]. The regularisation parameter α serves to trade balance the contribution of the neighbour score and the initial score.

PRINCE boosts its performance by removing predictions which fail to produce a score higher than 0.021 on the entire network after the diffusion. The improvement reported nearly doubles the amount of genes predicted in rank 1, while reducing the recall by 74% [206].

Finally, PRINCE shows an application to predict protein complexes, where each complex is produced taking all genes in the network up to a diffused score t. If less than 20 proteins are obtained, the neighbour of the current putative complex with the highest score is added iteratively until 20 are collected.

The putative complexes C are then reduced in size iteratively by removing the protein which causes the biggest improvement in the likelihood function $L(C)$. The process continues until no protein can be removed to improve the likelihood. This function evaluates the connectivity among the proteins included in the complex, and is defined as:

$$L(C) = \sum_{(u,v) \in E(C)} \log \frac{\beta}{w(u,v)} + \sum_{u,v \in V(C), (u,v) \notin E(C)} \log \frac{1-\beta}{1-w(u,v)},$$

where $V(C)$ and $E(C)$ are all the vertices and edges for complex C, and the parameter $\beta = 0.9$ is related to the sensitivity to the connections [206].

3.3 ProDiGe

ProDiGe [135] is a family of kernel-based disease gene prediction methods which rank all genes within the protein-protein interaction network for a given diseases. ProDiGe learns missing disease-gene associations using a one-class SVM, where known associations are seeded as positive labels and the other associations are unlabelled. The model considers all disease-gene associations at once – i.e. points are disease-gene tuples $\langle D, G \rangle$. To cope with points in the disease-gene space, the authors define the SVM kernel as a product between a disease kernel $K_{disease}$ (which

establishes the pairs of diseases that share targets through a disease similarity matrix) and a gene kernel K_{gene} (which establishes a similarity among the genes). Formally, the kernel between two disease gene pairs is defined as [135]:

$$K_{pair}\left((D,G),(D',G')\right) = K_{disease}(D,D') \times K_{gene}(G,G').$$

While the approach supports multiple gene kernels, the PPI kernel from the publication is shown to yield the best results among the first couple hundred predictions [135] – i.e. K_{gene} is the symmetric Laplacian of the adjacency matrix of a PPI network (see Section 2.3.1).

The four methods in the family (ProDiGe1 to 4) differ in the disease kernel used: ProDiGe1 does not share genes (i.e. $K_{disease} = I$ is the identity matrix), ProDiGe2 establishes an uniform low probability to genes from other diseases (i.e. $K_{disease} = 1 + I$ is the identity plus a constant), ProDiGe3 allows genes to be shared by using a disease similarity matrix (i.e. $K_{disease} = V$ is the van Driel similarity matrix [205]), and ProDiGe4 adds the kernels from ProDiGe1 and ProDiGe3 to give additional weight to the genes coming from the disease of interest (i.e. $K_{disease} = I + V$).

3.4 CATAPULT

CATAPULT (Combining dATa Across species using Positive-Unlabeled Learning Techniques) is another SVM based method, which considers a one-class classification problem to rank missing disease gene associations [186].

CATAPULT is designed to consider that the absence of information does not imply negative information, and incorporates disease gene associations from multiple species to provide more information. This produces a kernel composed of heterogeneous data sources C, defined as:

$$C = \begin{bmatrix} G & P \\ P^T & Q \end{bmatrix},$$

where G is a gene kernel, Q is a disease kernel, and P is a disease-gene association matrix. The binary matrix G of size $g \times g$ is produced by combining edges from HPRD [160] and HumanNet [103]. The binary matrix P of size $g \times (p_1 + p_2 + \cdots + p_n)$ is used to group gene associations from different species as $P = [P_1 \;\; P_2 \;\; \ldots \;\; P_n]$, where P_i is the disease gene association matrix for species i. The block diagonal matrix Q of size $(p_1 + p_2 + \cdots + p_n) \times (p_1 + p_2 + \cdots + p_n)$ contains diseases-disease similarities for the different species. However, the only non-zero block corresponds to human and contains the van Driel disease similarity [205].

This kernel considers elements as disease-gene tuples, and is trained using bagging [14] (for 30 bootstraps) to reduce the variance in the classifier. Furthermore, it differentiates the penalties for false positive errors (low penalty) and false negatives (high penalty), to account for the small amount of known gene associations in comparison to the unlabelled data. The final score is given by the average distance of the unlabelled point to the 30 trained bootstrap hyperplanes.

3.5 DIAMOnD

DIAMOnD [55] is a recent disease module prediction method based on direct neighbour analysis which starts from a set of initial seeds and iteratively increases the module by adding new genes. At each iteration, the algorithm evaluates which genes have more connections to the existing disease module than expected by random chance, using the hypergeometric distribution as the null model. Under the model, the probability of a randomly selected gene with with k interactors to be connected to k_s seeds is:

$$p(k,k_s) = \frac{\binom{s}{k_s}\binom{N-s}{k-k_s}}{\binom{N}{k}},$$

where N is the number of vertices in the network, and s is the number of genes in the current disease module. The most significantly connected gene according to this model is then added and the authors consider the first 200 to 500 genes as the recovered disease module. The significance of the connections is given by:

$$significance(k,k_s) = \sum_{k_i=k_s}^{k} p(k,k_i),$$

where the significance can be seen as the p-value of observing k_s or more connections to the seeds. Notice that the set of genes in the module grows by one per iteration, and the significance is recalculated on each step.

3.6 Discussion

Disease gene prediction comes frequently as a prioritisation procedure, in which all genes within a set (or graph) are sorted according to their likelihood of association to a disease. Beyond this generalisation, there is scarce standardisation among authors, which use different input data and processing methods.

Endeavour can be considered a staple ensemble method. While any given ranking presented in this algorithm could be considered of particular high quality, their

combination provides confirmation of non spurious results. However, network based methods proved to produce better results than this approach. A particular drawback is that all training is independent, and no lateral information is given between the instances.

PRINCE is likely to be the most well known disease gene prediction method. The usage of phenotype similarity to share information between diseases and the conservation of prior information constituted key developments of this method. A major drawback for is usage is the lack of an independent executable. Currently, PRINCE is only available as a Cytoscape plugin called PRINCIPLE [58]. Its usage requires manual commands and its output is limited to 100 results.

ProDiGe vastly outperforms PRINCE, and is able to integrate multiple sources to produce gene-gene associations. Furthermore, it provides several versions that are suitable on different gene recall thresholds. CATAPULT, on the other hand, contains multiple disease-gene association sources that come from species other than human. While CATAPULT outperforms ProDiGe4 on the first 100 results on several test configurations [186], its performance is still comparable to ProDiGe1.

While DIAMOnD is a relatively simple algorithm, it is the only method designed to retrieve disease modules based on a existing network. The alternative, DiME (Disease Module Extraction) is a method proposed by Liu *et al.* [113] that recovers disease modules through the analysis of gene expression. The goal of the method is to build a co-expression network between significantly expressed genes, which shows some desired topological properties (a significant B-score [101] and power-law degree distribution [10]). This approach does not consider PPI networks neither in the module construction, or the validation phase. DiME seems to strive for different goals in the construction of disease modules, and a straightforward comparison appears hard to produce while considering the evaluations available in the DiME publication.

ProDiGe1 and ProDiGe4 are chosen as representatives of the disease gene prediction methods as they have been shown to outperform other well known methods, such as PRINCE [206], Endeavour [3], and a multiple kernel learning approach (MKL1class) [135] in the top 200 predictions. Additionally, DIAMOnD is used as the reference disease module prediction method, and as a disease gene prediction method for completeness. Although DIAMOnD is not intended to be a fully-fledged disease gene prediction method, the order in which the genes are added to the module naturally produces a ranking that prioritises disease genes.

Chapter 4

Experimental setup

This Chapter formally defines the disease gene prediction and disease module prediction tasks as used in this work. Then, it describes the evaluation measures and datasets used for each problem.

4.1 Problem definition

Without loss of generality, the output of a network based gene prioritization method can be seen as a ranking. If any method produces ties within their ranking, they can be thought to produce a non-deterministic ranking: tied elements will be sorted randomly, where each element has a uniform probability to end at one of the tied positions.

The input of a prediction is defined by a disease of interest (or *query*) and the known gene associations. While different networks, data sources or parameters might be used, those are considered fixed to simplify the discussion. The only important consideration is that the prediction output only includes genes not known to be associated to the query.

Given a query and the known gene associations, the prediction output is expected to rank the yet unknown gene associations as high as possible. Although in real situations the number of missing genes might be unknown, testable cases contain one or more expected targets. This work adds a further constraint to limit the evaluation up to the first 200 predictions, this threshold is comparable to other proposals of the field [55, 135, 186, 206, 231] and represents a reasonable limit for experimental validation.

The basic case is the prediction of a single target, where the performance of the prediction is given by the target rank. Naturally, an ideal prediction ranks the target in position 1. Experiments with a single target are the most frequently used

in the literature [213], and the evaluation of a method quantifies the performance over multiple predictions. In this work, a set of single target predictions is evaluated by the percentage of targets found upon a number of predictions retrieved (*recall at threshold*, presented in Section 4.2.1).

However, the disease module prediction intends to identify multiple genes simultaneously. The ideal prediction has all k targets ranked in the first k positions. This work proposes the usage of a ROC curve normalised for the first 200 false positive predictions (presented in Section 4.2.1) to evaluate the prediction of disease modules.

While the performance of disease module predictions can be quantified by the normalised ROC, an in-depth validation the modules involves a qualitative analysis of topological and biological properties which are presented in Section 4.2.2.

Experiments

This work classifies the experiments based on the number of known genes and the source of the targets, which offers well-defined scenarios. Queries with 1 or more known genes are *charted*, while queries no known genes are *uncharted*. Gene targets can be obtained by mining a gene association database in different points in time; the gene associations added to each disease when comparing the old database to the new database are used for the *time-lapse* experiments. Additionally, gene associations can be removed from a disease in order to be predicted back as targets. These constitute the *synthetic* experiments.

Furthermore, these synthetic targets can be produced for charted diseases by removing a single gene at a time (i.e. *leave-one-out*) or by removing a percentage (used for disease *modules*). The disease module experiments analyse ability of the prediction to reconstruct different percentages of the module (or the performance trend when keeping from 0% to 95% of the module as seeds). These experiments include 10 uniform random samples of target genes for every percentage in order to reduce sampling bias and excessive costs in module testing.

4.2 Evaluation

4.2.1 Quantitative evaluation measures

The following measures are used to quantify the performance of a prediction given as a ranking. The recall at threshold is used to evaluate a single target at a time, while the normalised ROC can be used to evaluate one or multiple targets.

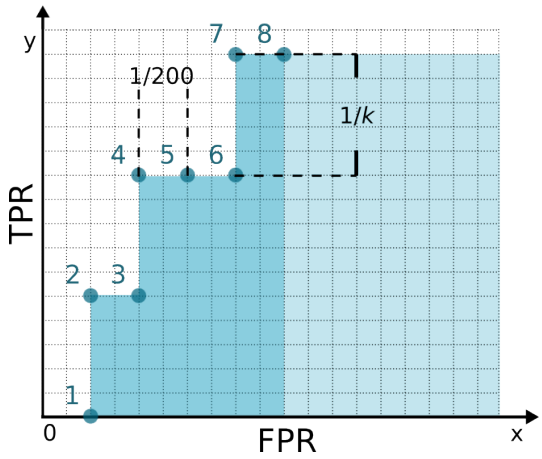


Fig. 4.1 **Example ROC curve normalised to X false positives for a discrete prediction sequence.** This example shows a partial ROC curve produced for $\eta = 8$ elements, in a sequence with a total of m targets and X false positives. In particular, it shows the case where the first three targets are ranked in positions 2, 4 and 7.

Recall at threshold

The measure quantifies the performance of n predictions, where each prediction produced rank r_i for target i. The evaluation consists in the percentage of targets found within the top predictions up to a threshold τ, in particular the top 1, 10, 100 and 200 results. This is, the recall R_τ at the top τ is given as:

$$R_\tau = \frac{\#r_i \leq \tau}{n},$$

where $\#r_i \leq \tau$ is the number of r_i ranked τ or higher.

Notice that the different R_τ are not combined, and are given as the output. Furthermore, the vector $R = [R_1, R_2, \ldots, R_n]$ produces a ROC curve if normalised in the x-axis.

Normalised ROC

The measure quantifies the performance of a single prediction for k targets, where target i is ranked in position r_i. The measure is the area under a ROC curve trimmed and normalised for the first 200 false positive predictions. The ROC curve is defined as the set of points given by true positive rate (TPR in the y-axis) at a false positive rate (FPR in the x-axis), by changing the FPR. The following formulation produces the curve from the ranks r_i (Figure 4.1).

Without loss of generality, let $r_1 < r_2 < \cdots < r_k$ (lower is better). The TPR up to rank η is given by:

$$TPR_\eta = \frac{\#r_i \leq \eta}{k},$$

where $\#r_i \leq \eta$ represents the number true positives, and k represents the total number of positives. Analogously, the FPR up to rank η is given by:

$$FPR_\eta = \frac{\eta - \#r_i \leq \eta}{200},$$

where $\eta - \#r_i \leq \eta$ represents the number of false positives, and 200 is the total number of false elements to be considered. The normalised ROC curve is produced by tracing the (FPR,TPR) points obtained by increasing η from 1 until the FPR reaches 1 (i.e. until $\eta - \#r_i \leq \eta = 200$).

Two main alternatives to this measure seem intuitive: considering a fixed number of predictions (instead of a variable number of predictions up to 200 false positives) for the ROC curve, or a Precision-Recall (PR) curve. However, both of them may lead to unintuitive or undesirable results:

- The number of elements of the curve is not predefined because it makes predictions with different number of targets comparable. The usage of a fixed amount of predictions (e.g. 200) can produce questionable results as illustrated with a trivial example: a prediction with 198 targets ranked from 2 to 199 would be exceptional and would produce an AUC of 0.5, while a notably inferior prediction for 2 targets ranked 99 and 100 would also produce an AUC of 0.5. The current proposal gives an AUC of 0.995 to the former, and 0.505 to the latter prediction, which appears more appropriate.

- The AUPR is an interesting measure to compare pairs of predictions with the same number of missing targets, as it naturally assigns higher weights to the top predictions. However, there are some weird effects when the predictions do not have the same number of targets in every prediction. For instance, the evaluation of a method for an increasing number of missing known genes may yield[1]: $A_{PR}([1,200]) = 0.505$, $A_{PR}([1,50,200]) = 0.352$ and $A_{PR}([1,10,50,200]) = 0.32$, where the average position of the targets gets better, but the AUPR gets lower. The AUPR measure could make the analysis for trends in performance for different percentages of missing modules hard.

[1] A_{PR} represents the AUPR function, and the numbers within the brackets represent the ranks in which the targets are found.

4.2.2 Qualitative module evaluation

This work also discusses the evaluation of modules by a direct analysis of network properties and the coherence of biological functions. In particular, the correspondence of network features of putative modules compared to a gold standard. This case considers that the putative modules contain the gold standard, and the measures are related to the production of a good matching.

A qualitative analysis is also used for a smaller exploratory analysis on drug targets (Section 8.4). Notice that drug targets are those proteins that bind or interact with drugs and have their activity modified, so this is not an evaluation of performance of drug target sets. The analysis intends to determine if drug targets behave as *drug modules*, so the topological features of drug targets are compared to those found in disease modules. Additionally, it explores how the drug modules overlap with corresponding disease modules.

Correspondence to a gold standard

The direct evaluation of disease modules involve the validation of the network properties in comparison to a gold standard. The gold standard is composed of all diseases within OMIM with 2 or more known genes.

The baseline comparison is a set of random modules is produced by removing all genes from the gold standard diseases, and populating them from the HPRD genes with a uniform distribution (each module draws the genes without repetition, but genes can be repeated in different modules). Following this procedure, the number of modules and their sizes are preserved. The evaluation of random samples is the average of 1,000 sets of random disease modules.

The method's predictions are produced by creating synthetically uncharted diseases and then predicting the entire module. The set of synthetically uncharted diseases is obtained by taking each disease from the gold standard individually, and removing all their known gene associations. Notice that while a disease is made uncharted, all other known gene associations from OMIM are kept to allow the prediction of the modules.

The *Closeness* in a network module is measured by the average distance between all vertex pairs (not necessarily connected). However, the disease modules are frequently disconnected [127]. To avoid considering infinity in the calculations, the distance between vertices in different connected components is set as the diameter of HPRD (the source of the module graph) plus 10.

Separation, *Accuracy* and the *Jaccard coefficient* are measures calculated between pairs of modules, one of them is the gold standard, and the other is the module to be evaluated. The Jaccard coefficient is defined as:

$$J(S, GS) = \frac{|S \cap GS|}{|S \cup GS|}, \tag{4.1}$$

where S is the set of proteins being evaluated, and GS is the set of proteins of the corresponding gold standard.

Let n and m be the number of gold standard and predicted disease modules, and $\{n_1, \ldots, n_n\}$ and $\{m_1, \ldots, m_m\}$ the numbers of proteins of each of those modules. Let the confusion matrix be $T_{n \times m}$, where the elements $t_{i,j}$ represent the number of proteins common to the gold standard module i and the predicted module j. The separation Sep indicates the correspondence between the modules (ranging from 0 or none, to 1 or perfect), and is defined as:

$$Sep = \frac{F_{\text{row}} \cdot F_{\text{col}}}{\|F_{\text{row}}\| \|F_{\text{col}}\|}, \tag{4.2}$$

where F_{row} and F_{col} are the marginal sums per row and column of the confusion matrix T. The cosine distance between the marginal sums represents how the elements from the gold standard clusters are distributed in the predicted clusters. If every element predicted the correct cluster, the marginal sums would be equal $F_{\text{row}} = F_{\text{col}}$, and the cosine distance would equal to 1. Other distributions would cause it to decrease.

The accuracy relates the sensitivity and positive predictive value to penalise trivial cases of module matching [16]. The Sensitivity Sn is defined as:

$$Sn = \frac{\sum_{i=1}^{n} \max\{t_{i,j} | 1 < j < m\}}{\sum_{i=1}^{n} n_i}, \tag{4.3}$$

and the Positive Predictive Value PPV is defined as:

$$PPV = \frac{\sum_{i=1}^{n} \max\{t_{i,j} | 1 < j < m\}}{\sum_{j=1} \sum_{i=1}^{n} t_{i,j}}. \tag{4.4}$$

Then, the Accuracy Acc is defined as:

$$Acc = \sqrt{Sn \times PPV} \tag{4.5}$$

The *functional similarity* between proteins by measuring the pairwise Gene Ontology semantic similarity by Yang *et al.* [225], calculated using GOssTo [19] with the August 2017 release of the Gene Ontology and Human GAF annotation

files [31]. The GeneMANIA evaluation uses the March 2017 release of the dataset [136].

Assessment of modular features

The modular features are similarity in distribution of physical protein distances of modules and function specificity of a module.

The *physical distance* between protein targets is the minimum path length between a pair of proteins in an unweighted and undirected graph, that represents the HPRD PPI network. The distance distribution within different modules is later compared using Welch's t-test, defined as:

$$t = \frac{\bar{A} - \bar{B}}{\sqrt{\frac{\sigma_A^2}{|A|} + \frac{\sigma_B^2}{|B|}}},$$

where A and B are the sets of all pairwise protein distances within two modules, $\bar{X}$ is the mean of set X, σ_X^2 is the variance and $|X|$ is the number of elements.

The overrepresented categories per protein are obtained using Fisher's exact test [87]. Then, the pairwise functional similarity for a protein set is calculated using the Yang *et al.* semantic similarity [225] as in the previous evaluation.

The function of a drug module can be thought as the aggregation of all individual drug target functions. This analysis proposes to quantify the *function specificity* of a drug module by comparing the pairwise semantic similarities from the module targets to a global expected similarity. The global drug target similarity is modelled by approximating a distribution of all pairwise semantic similarities between drug targets. This baseline distribution is determined by the best fitting random variable distribution found in the Python SciPY 0.14.1 version.

Exhaustive testing determined the usage of the half logistic distribution as the model for the cumulative semantic similarity distribution [18]. This is:

$$F(k) = \frac{1 - e^{-k}}{1 + e^{-k}},$$

where the k value is scaled by the real values a and b:

$$k = (x - a)/b,$$

and x is a similarity within the set.

Following this model, a drug module is considered *specific* if the mean pairwise similarities is found above the 80 percentile of the baseline distribution.

4.3 Data sources

4.3.1 Disease Gene Associations

OMIM

The main source for disease gene association data for this work are OMIM databases [125]. Two versions of the database (2013 and 2017) were mined to allow *time-lapse* experiments. These experiments test diseases existing in the 2013 database which have gained genes in the 2017 database. Table 4.1 contains general information about the diseases covered on each OMIM database.

Table 4.1 **Relevant counts from the OMIM Databases.** Only diseases with 2 or more known genes can be used for synthetic *leave-one-out* experiments. The number of uncharted diseases accounts for all diseases with no known molecular basis which have annotated publications (diseases with annotated suspected genes are not included in this count). The number of unique disease genes is shown as some genes belong to multiple diseases. The number of disease gene associations are all unique disease-gene pairs annotated in the OMIM database.

	OMIM Release	
	2013	2017
Diseases with 1 or more known genes	4,870	5,992
Diseases with 2 or more known genes	293	264
Uncharted Diseases	2,670	2,388
Unique Genes	4,040	4,820
Disease Gene Associations	6,303	7,292

The OMIM database is a hand curated collection of experimentally verified data, so the difference between versions is generally due to the curation of new scientific publications found between the releases. There are 289 diseases that were uncharted in the 2013 OMIM database which are charted in 2017. Some of them acquired more than one gene, which amounts to a total of 292 disease-gene associations available for the uncharted *time-lapse* experiments. Likewise, there are 66 charted diseases in 2013 which have additional genes in 2017, for a total of 95 additional disease-gene associations available for the charted *time-lapse* experiments.

Notice that while the 2017 OMIM database includes 989 more disease associations than the 2013 release, over 1,400 associations are not included in the older version. This difference comes from changes in diseases identifiers between versions,

changes in the genes associated to the diseases, and diseases removed from the database.

ClinVar

Additionally, this work uses the ClinVar database [102] that contains protein variants associated to each phenotype. Only the 2018 version of the ClinVar database is found available, therefore no *time-lapse* experiments are performed using this database.

Table 4.2 **Relevant counts from the ClinVar database.** Only diseases with 2 or more known genes can be used for synthetic *leave-one-out* experiments. The number of unique disease genes is shown as some genes belong to multiple diseases. The number of disease variants counts all disease to protein variants annotated in the database (some proteins have multiple SNPs annotated a single disease).

	ClinVar
Diseases with 1 or more known genes	3,607
Diseases with 2 or more known genes	133
Unique Genes	2,758
Disease Variant Associations	32,359

ClinVar is focused on the annotation of human variants, and includes additional polymorphisims not associated to disease. Table 4.2 summarises all database entries considered for evaluation in this work. These entries are all those associated to an OMIM identifier.

DIAMOnD disease modules

Ghiassian *et al.* provide manually curated disease modules for 70 complex diseases [55]. The dataset contains 1,536 different genes, where some are associated with more than one disease, for a total of 2,843 disease-gene associations.

This work mapping these diseases to OMIM identifiers by matching disease names and descriptions found in OMIM. The full mapping is presented in Appendix B.1.

4.3.2 Interactomes

Several freely available protein protein interaction networks are used to show the general applicability of the methods proposed in this work. These include weighted

and binary networks with both experimental and predicted data: HPRD [160], DiamondNet [55] and BioGRID [24] are binary experimental networks; HIPPIE [175] is a weighted experimental network; FUNCOUP [178] is a large weighted network including both experimental and predicted data. Table 4.3 presents an overview of these networks, and their coverage of genes found in the OMIM databases (seen in Table 4.1).

Table 4.3 **Characteristics of protein-protein interaction networks.** Coverage shows the fraction of the different disease genes from the OMIM database found in the network (see Table 4.1). Evidence describes if the edges are obtained through experimental validation (exp) or inference (inf). Edge type indicates whether the edges are binary or weighted.

	Network				
	HPRD	*DiamondNet*	*BioGRID*	*HIPPIE*	*FUNCOUP*
Nodes	9,670	13,460	19,803	16,552	18,113
Coverage OMIM2013	54%	65%	69%	71%	71%
Coverage OMIM2017	54%	66%	71%	74%	74%
Edges	39,220	141,296	279,187	239,684	4,476,818
Evidence	Exp.	Exp. + Inf.	Exp.	Exp.	Exp. + Inf.
Edge Type	Binary	Binary	Binary	Weighted	Weighted

Note that even if a gene is found in a network some methods can still fail to predict it. Both ProDiGe and DIAMOnD can only find genes that are found in the same connected component as the known genes from the query disease. In practice, the prediction is limited to targets found in the main connected component. However, Cardigan constructs a vector of initial seeds with every known disease gene. Therefore, it can predict targets in any connected component that has a known disease gene from OMIM.

Besides these PPI networks, this work analyses the usage of an interactome which includes protein interactions through interfaces. The information used here is provided by the Interactome INSIDER [130], a recently available resource. This database contains information about the residues which are predicted (with a method called ECLAIR) to be in the interaction interface. This information can be used to build a standard PPI network, and this work proposes a construction for a second

interface-based network which is presented in Section 7.1.1. The procedure creates a network where the nodes are interfaces, and the edges represent the interaction between interfaces. Table 4.4 presents a summary of both networks built from Interactome INSIDER.

Table 4.4 **Summary of the networks derived from Interactome INSIDER.** PPI references to a standard protein-protein interaction network built from the database, and ECLAIR references the interface-interface network proposed in Section 7.1.1. Coverage shows the fraction of the different disease genes from ClinVar found in the network (see Table 4.2). LCC stands for the Largest Connected Component.

	Network	
	PPI	*ECLAIR*
Nodes	15,046	17,184
Edges	238,710	238,710
Coverage ClinVar	84%	84%
Connected components	304	477
Proteins in LCC	14,708	14,692
Nodes in LCC	14,708	16,619
Nodes outside the LCC	338	565

Notice that ECLAIR predictions come with a confidence score, and residues with low confidence scores are not predicted to be in the interface. This yields some interactions where an interface contains no residues, which are referred to as ∅ interfaces in this work. Therefore, Table 4.5 presents details of the proteins included in the INSIDER ECLAIR network.

Table 4.5 **Details of the proteins included the INSIDER ECLAIR network.** Each row counts the number of proteins found in the network with a given characteristic. Notice that a protein yields a node per interface in the ECLAIR network.

	Total	*Percent*
1 interface	13,078	87%
2 or more interfaces	1,968	13%
contain a ∅ interface	7,062	47%
contain a ∅ and other interface	1613	11%

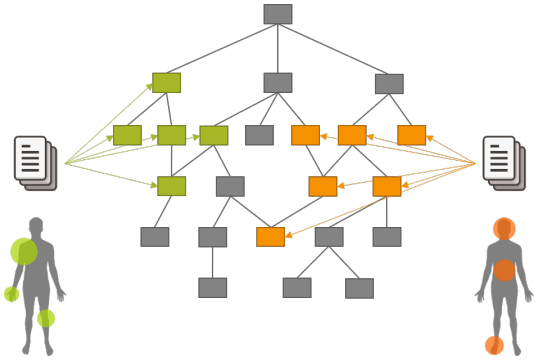


Fig. 4.2 **Phenotype characterisation of a disease - Caniza similarity.** The first step is to gather PubMed publications describing the diseases (green and orange). Then, MeSH terms are extracted from those documents, and are annotated onto the MeSH ontology. Finally, the similarity is quantified as an information content based distance in the ontology.

4.3.3 The Caniza disease similarity

Caniza *et al.* [20] proposed to characterise a disease phenotype by collecting MeSH terms of relevant scientific publications about the disease, and annotating them in the MeSH ontology (Figure 4.2). The phenotype similarity for a pair of diseases is quantified through the information content of the lowest common ancestors between the sets of the annotated terms.

Following the reasoning by Caniza *et al.*, a term with few annotated diseases tells a lot about those diseases, while very common terms do not aid in telling diseases apart. Therefore, the importance of the term is defined by its information content. On the field, the information content of a term $IC(t)$ is commonly quantified by the negative log-likelihood [180]:

$$IC(t) = -log(p(t)),$$

where $p(t)$ is the probability of a disease being annotated on term t.

Then, the similarity between a pair of terms t_i and t_j is calculated by the information content of their most informative common ancestor in the ontology [167]. The common ancestor represents the conceptual similarity between the terms. Formally:

$$sim(t_i,t_j) = \max_{c \in C(t_i,t_j)} IC(c),$$

where $C(t_i,t_j)$ is the set of common ancestors to both t_i and t_j in the ontology.

Since diseases are annotated under multiple terms, the similarity between a pair of diseases is the maximum similarity between all pairs of terms across the diseases. Formally, if diseases d_a and d_b are annotated with terms $a_i \in A$ and $b_i \in B$ respectively, the Caniza similarity between the diseases $S(d_a, d_b)$ is calculated as:

$$S(d_a, d_b) = \max_{a_i \in A,\, b_j \in B} sim(a_i, b_j).$$

The Caniza similarity is provided as a symmetric square matrix where the element i, j represents the similarity between disease i and disease j.

The Caniza similarity considers that two diseases will have a high similarity if they share any phenotype, even if the aggregation of phenotypes makes them different. Notice that the similarity is not an Euclidean distance, so it is not transitive. If a disease is similar to two diseases, the two other diseases are not necessarily similar ($A \sim B \wedge A \sim C \not\rightarrow B \sim C$).

The similarity is calculated on the MeSH ontologies *Anatomy* [A], *Diseases* [C], *Chemicals and Drugs* [D], *Analytical, Diagnostic and Therapeutic Techniques and Equipment* [E] and *Phenomena and Processes* [G], combined with an extra root node which connects all five ontologies [20].

This work uses the Caniza similarity calculated on the 2013 OMIM database, and the 2017 OMIM database. Notice that the OMIM database holds records of curated publications associated to each disease as PubMed identifiers [125].

Chapter 5

Mapping disease modules from phenotype

This Chapter focuses on the usage of phenotype similarity for the characterisation of disease modules, and its relation to the prediction of disease genes. The characterisation is given as a disease module prediction method called *Ordinal*[1]. Earlier works, such as the proposals by Li *et al.* [107], Vanunu *et al.* [206] and Mordelet *et al.* [135], have exploited a human phenome quantification to guide random walks in graphs, diffusion processes, and inductive semi-supervised models, respectively, to prioritise disease genes [205]. However, scant research approaches the general relation between the modular nature of phenotypes across multiple diseases [9, 55, 113].

5.1 The Ordinal method

Ordinal is a disease phenotype approach to predict disease modules for uncharted diseases using known disease genes. The idea is based on the fact that a high phenotypical similarity between diseases correlates to closeness in their genotype [20]. The predicted modules are directly validated through topological properties within a PPI network, and indirectly by considering the method within a disease gene prediction context. The latter validation serves a further purpose, to quantify the possible gains by including topological completion of the network modules upon these phenotypical modules.

Ordinal consists of three steps to construct the disease modules: 1) collect a list of highly similar contributor diseases from which to transfer information; 2) all

[1]Ordinal is a yet unpublished method produced in collaboration with Horacio Caniza under the supervision of Prof. Alberto Paccanaro. While the base ideas for method were developed by all the authors, the code implementation, evaluation for disease gene prediction, and the discussion presented here are my own.

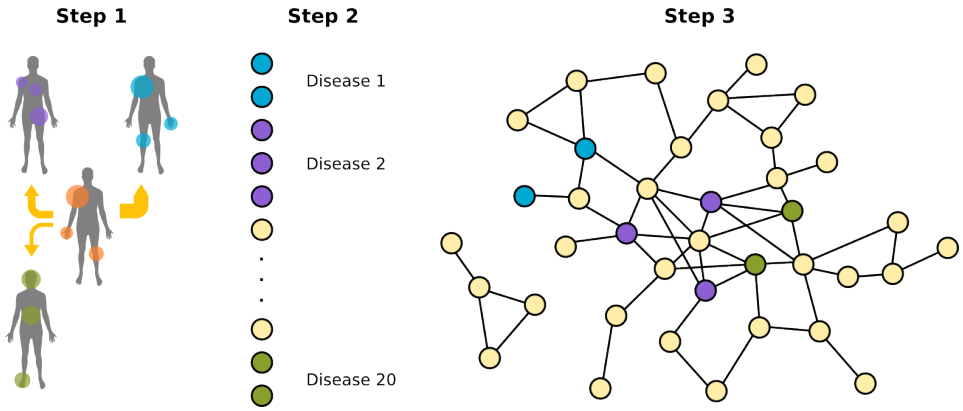


Fig. 5.1 **The Ordinal Process.** Given an uncharted disease in OMIM (orange), a set of the 20 most similar charted diseases is collected – i.e. the contributor diseases (Step 1.). The known molecular basis of the contributor diseases are extracted and become the genes of the putative module (Step 2.). The set of interactions is extracted from HPRD, their interactions are assembled into a disease module for the uncharted disease (Step 3).

known molecular basis are extracted from these diseases, which produce the set of putative proteins; 3) all interactions between the set of putative genes are mined from HPRD [160], which is a binary PPI network composed of high quality Y2H experiments. Figure 5.1 shows a sketch of the entire Ordinal pipeline.

5.2 Results

While multiple authors develop pipelines to identify modules of particular diseases [150, 51], the only general computational method that recovers disease modules is DIAMOnD [55]. However DIAMOnD requires some known molecular basis to produce a prediction (see Section 3.5 for further details). A literature review revealed no methods aimed to the construction of modules for uncharted diseases, rendering a direct comparison to other methods impossible (arguments regarding DiME [113] are discussed in Section 3.6). Therefore, the primary evaluation consists in comparing the predicted modules to a gold standard and random modules using the qualitative module measures of functional similarity (semantic similarity), closeness (intra-module distances), separation, accuracy and the Jaccard coefficient. Section 4.2.2 describes the construction of the gold standard (high quality disease modules), the random sets (baseline proteins found in the network), and the measures.

The first evaluation procedure consists in the observation of the essential *guilt-by-association* principle [10]. In this context, the predicted modules must show a function similarity and maintain a topologically compact module in the interactome

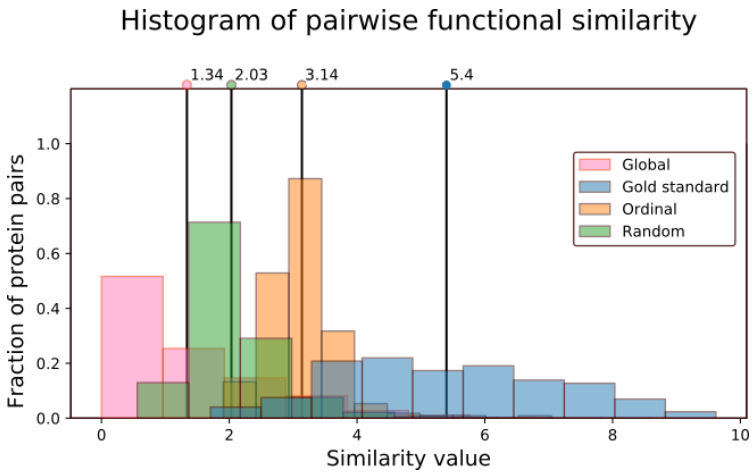


Fig. 5.2 **Histograms of semantic similarity values on the PPI network.** Green bars represent the average on the entire HPRD PPI network. Blue bars show the pairwise semantic similarity of the proteins in random disease modules and magenta bars the distribution in the modules predicted by Ordinal. Finally, red bars show the distribution of pairwise semantic similarity within the Gold Standard disease modules in OMIM. The solid vertical lines show the average of each distribution, with the mean value indicated above.

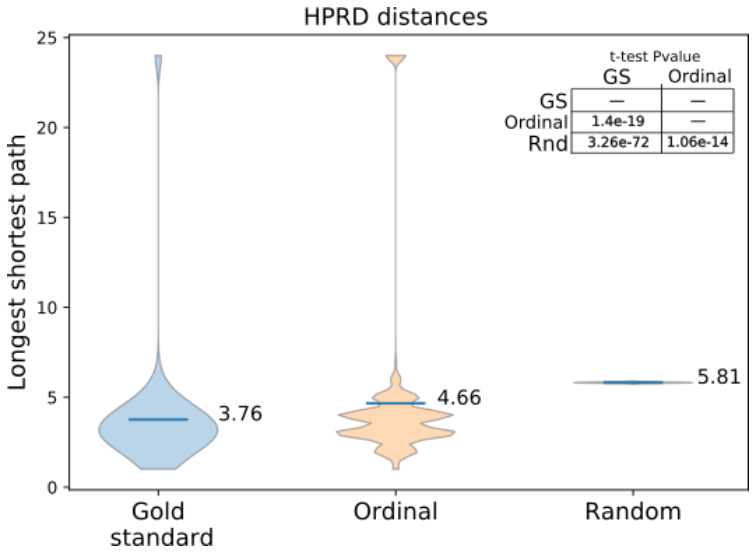


Fig. 5.3 **Intra-module distance for Gold-standard, Ordinal and random sets.** The Y-axis shows the average intra-module distance between proteins in the corresponding modules. The table at the top shows the pair-wise t-test p-values between the sets. The distance between disconnected proteins is defined as the diameter of HPRD plus 10

[164, 55, 140]. Figure 5.2 shows the function similarity between all proteins in the module through the comparison of the annotated GO terms. Figure 5.3 shows the closeness between the proteins within the predicted modules.

This initial evaluation shows that Ordinal indeed produces modules which are significantly different from random modules, and approximate the distribution of the gold standard disease modules. This test is necessary as there are no guaranties that the closeness in phenotype and genotype will hold for the set of diseases collected following the Ordinal algorithm.

While these properties are essential for a disease module, Ghiassian *et al.* show that topological modules on the interactome are not able to fully capture disease modules [55]. Table 5.1 shows a benchmark for the produced modules using the Jaccard coefficient, Separation and Accuracy with respect to its corresponding gold standard module [16]. The Jaccard coefficient measures similarity between sets, separation measures how well a given putative module is separated from the other modules and accuracy measures how well a putative module corresponds to its corresponding gold standard module.

Table 5.1 **Comparison of module quality.** The table compares the quality of Ordinal's predictions, measured in terms of Separation, Accuracy and the Jaccard coefficient, to the quality of randomly generated modules. All scores range from 0 to 1, and higher values are better. We create a composite score summing the individual values

	Module source	
	Predicted	Random
Separation	0.0195	0.00767
Accuracy	0.2979	0.079
Jaccard	0.0238	0.0003
Composite	0.3413	0.087

Finally, Ordinal can also be considered as a fully blown disease gene prediction method, by ranking the genes of the contributor diseases in order of their similarity. However, since Ordinal is not able to distinguish between the set of proteins it brings from a single disease, they are assigned a priority at random, and 100 samples are generated to produce the results. Figure 5.4 shows that while Ordinal's design fails to identify genes between the top 10 predictions, it contains sufficient module identification power to outperform dedicated methods in the top 200 results. The

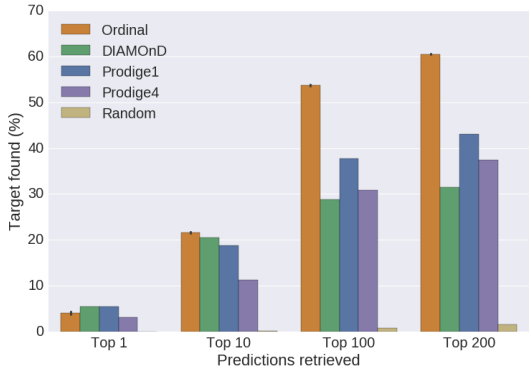


Fig. 5.4 **Disease gene prediction for charted diseases in a leave-one-out procedure – average recall.** The chart shows a comparison between Ordinal, DIAMOnD and ProDiGe for disease gene prediction on diseases synthetically missing one gene (for more details see Section 4.1), on HPRD. The bars show the percentage of targets found within the first 1, 10, 100 and 200. predictions.

selection of DIAMOnD [55] ProDiGe [135] as methods of reference are further explained in Section 3.

5.3 Discussion

The *guilt-by-association* principle established that a closeness in genotype was reflected in a closeness in phenotype within the protein level [10]. However concept of phenotype can be thought of in different abstraction levels. Ordinal, despite being a naive approach, clearly shows the intrinsic correlation between observable disease phenotypes and the *guilt-by-association* principle beyond the protein level.

Protein phenotype can be defined by the presence of structural properties (such as conserved domains, protein interaction interfaces, compound binding interfaces or motifs), or by the set of interacting proteins [36]. However, many computational approaches [167, 78, 110, 154, 32, 225] focused on the quantification of phenotype as a similarity from an ontological description of the known functions and structures of a protein, mainly by mining GO. The development of these approaches has repeatedly served as validation of the correlation between protein phenotype and genotype.

On the other hand, the definition of a disease phenotype originally derives from a natural macroscopic observation, and ontologies have been built since 1800 [147]. Here, the MeSH ontology [143] serves an analogous purpose to GO. van Driel *et al.* [205] and Caniza *et al.* [20] produced methods to quantify disease similarity within the MeSH ontology, validating the correlation between disease phenotype and genotype.

Since Ordinal manages the construction of protein modules based on disease phenotype, it shows that both correlations are simultaneously coherent. The disease modules produced by Ordinal are located in compact areas (Figure 5.3) in the interactome and contain proteins with coherent functions (Figure 5.2).

The application for disease gene prediction indicates the quality of the disease modules, and adds support to the idea that topological modules fail to characterise disease modules [55]. While other functionally related proteins arguably belong the module, disease genes are the core elements. Ordinal is able to retrieve more than 50% of the known disease genes within the module, and outperforms the other methods in this regard. Additionally, this application proves the qualitative improvement offered by the Caniza *et al.* similarity over the van Driel *et al.* similarity (used within ProDiGe4), and encourages its usage on a dedicated method.

Chapter 6

Charting disease gene associations

This Chapter presents my method named Cardigan (ChARting DIsease Gene AssociatioNs) that predicts genes for both charted and uncharted diseases, and can also predict disease modules. Cardigan is based on a semi-supervised algorithm that propagates labels on the interactome. These labels integrate disease phenotypic information expressed as a similarity measure between diseases, which is obtained by mining and comparing MeSH terms [143] relevant for each disease on the MeSH ontology.

A Python implementation of Cardigan is downloadable from the paper website http://www.pacanarolab.org/Cardigan. The code can be downloaded alone, or with a data bundle, which includes a precomputed Caniza similarity matrix, as well as parsers for the networks used in the evaluation (HPRD, DiamondNet, Biogrid, HIPPIE and FUNCOUP). A detailed manual for its usage is presented in Appendix A.

Section 6.2 presents thorough experimentation which shows that Cardigan outperforms state-of-the-art methods in disease gene and disease module prediction. Section 6.3 discusses the evaluation procedure, the main advantages of the proposal, and the applicability of the method in disease research.

6.1 The Cardigan algorithm

The idea is to exploit the fact that disease modules of diseases with a similar phenotype should be placed close-by on the interactome [10]. Therefore, when predicting disease genes for a specific disease, genes of phenotypically similar diseases should provide useful information.

To predict disease genes for a given disease (*query disease*), Cardigan begins by calculating its phenotypic similarity to every other disease in OMIM using the

approach developed by Caniza *et al.* [20]. Next, Cardigan assigns a weight to each known disease gene. The weight is related to the Caniza similarity between the query disease and the disease to which the gene is associated (Figure 6.1C). Weights of disease genes are real values between 0 and 1 and are calculated by rescaling the Caniza similarity through a sigmoid function (Figure 6.1B), whose parameters are learned (see 6.1.2). If a gene is associated with more than one disease, Cardigan uses the highest similarity value[1]. Genes which are already known to be associated with the query disease, if any, are assigned a weight equal to 1. For a given query disease, the set of weights assigned to the disease genes is henceforth called the *Query Weight Set* (QWS) for that disease. A detailed formulation of the QWS is presented in Section 6.1.1.

The next step is to propagate the QWS through the graph with a semi-supervised learning procedure (transition between C and D in Figure 6.1). Cardigan uses the consistency graph diffusion method from Zhou *et al.* [236]. This is a graph labelling procedure based on minimizing a cost function that takes into account network weights and an existing set of labels. Let a weighted PPI network with n nodes be represented as an adjacency matrix $W_{n\times n}$, where each element W_{ij} is the weight between genes i and j (if the network is binary, then all the values in W are binary, indicating the presence or the absence of an interaction). The final labelling vector F (of size n) having one element for each gene, whose value is related to the probability of that gene of being associated with the query disease, is obtained by minimizing the following cost function:

$$C(F) = \frac{1}{2}\left(\sum_{i,j=1}^{n} W_{ij}\left\|\frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}}\right\|_2^2 + \mu\sum_{i=1}^{n}\|F_i - Y_i\|_2^2\right), \tag{6.1}$$

where vector Y (of size n) is the QWS and $\mu > 0$ is a regularization parameter, and D is a diagonal matrix (of size n) whose elements are the row-wise sum of the weight matrix ($D_{ii} = \sum_{j=1}^{n} W_{ij}$).

The following analysis gives intuition for the method, and presents a closed form solution that minimises the cost function. The cost function being minimised is the sum of two terms. The first term accounts for the consistency of the labels of adjacent nodes (reflecting the *guilt-by-association* principle) – this term is minimized

[1]Roughly speaking, the different weights of the gene represent different phenotypes caused by the gene. Choosing the highest weight equates to consider only the phenotype caused by the gene most similar to the query phenotype. E.g. FTL is a known disease gene, and it is associated to iron storage. The excess or deficiency of this protein leads to different phenotypes. Then, the QWS for Sickle Cell Anemia will have a high weight for FTL, since it is associated to a phenotype of iron deficiency. It seems adequate to consider that FTL is likely to be close to the Sickle Cell Anemia genotype, even if it is associated to other phenotypes.

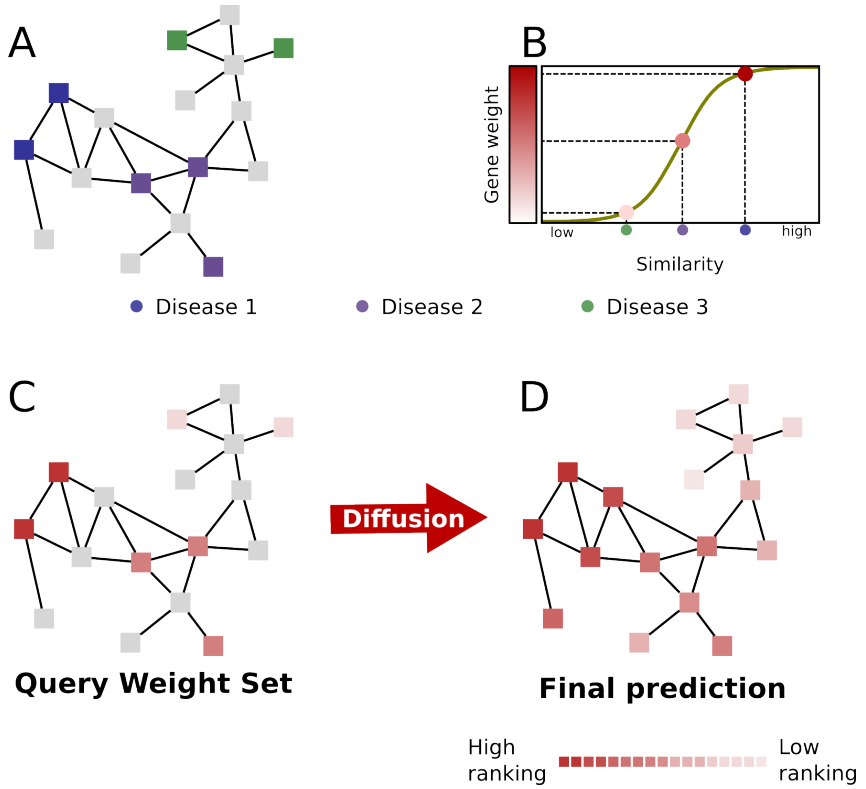


Fig. 6.1 **The prediction on an uncharted disease using Cardigan.** A) The PPI network with disease genes associated to 4 different diseases (red, green, purple, blue). B) The Caniza similarity is transformed to a weight. C) The query weight set (QWS) which serves as initial seed set for the diffusion process. D) Presents the final state of the network after the diffusion process. Notice how all genes have acquired a weight. These weights are used to rank all genes and constitute Cardigan's prediction.

when adjacent nodes have similar labels (i.e. the difference between F_i and F_j becomes small). Also note that the importance of the difference between F_i and F_j is proportional to the edge weight W_{ij} – i.e. it is related to the probability of the interaction. At the same time, the role of the second term is to conserve the initial labels (QWS), thus emphasise the reliability of the initial data for the prediction – this term is minimised when the nodes labels F_i are the same as the initial labels Y_i. Finally the μ parameter controls the relative importance of the two terms, while the D_{ii} terms serve as a normalisation parameters for the node degree. The vector F that minimizes the above cost function can be interpreted as a gene ranking (Figure 6.1D), and constitutes the output of Cardigan. In particular, Zhou *et al.* determined that the minimum of the cost function from Equation 6.1 has the following closed form solution:

$$F = \beta(I - \alpha S)^{-1} Y \tag{6.2}$$

where $S = D^{-1/2} W D^{-1/2}$, $\alpha = \frac{1}{1+\mu}$, and $\beta = \frac{\mu}{1+\mu}$.

It is important here to note that Cardigan is able to predict genes both for charted and uncharted diseases. In fact, the only input for the procedure is the QWS, which can be obtained for both groups of diseases. The only difference is that charted diseases will contain genes with label equal to one corresponding to disease genes already known for those diseases. Furthermore, the method can be used for the prediction of disease modules, since the top predictions of Cardigan can be interpreted as the disease module for the query disease.

6.1.1 The Query Weight Set

The QWS is a vector produced by weighting and combining known gene associations from OMIM. Weights of disease genes are real values between 0 and 1 and are calculated by rescaling the Caniza similarity through a sigmoid function. Genes which are already known to be associated with the query disease, if any, are assigned a weight equal to 1.

Formally, let the binary matrix G of size $m \times n$ represent the known disease gene associations, where the m rows represent diseases and the n columns represent genes – i.e. $M_{di} = 1$ represents that gene i is associated to disease d. The weighted matrix H of size $m \times m$ represents the Caniza similarity [20], where $H_{dd'}$ is the similarity between disease d and disease d'.

The Caniza similarity is a real positive number where increasing values represent increasing similarity. These are scaled by a sigmoid to obtain a bounded real number between 0 and 1. After scaling, diseases close to 1 are expected to be very similar to

the query and their genes are considered good candidates; on the contrary, diseases close to 0 are not expected to provide good candidates. Let a scalable sigmoid $\sigma_{s,c}$ be defined as:

$$\sigma_{s,c}(x) = \frac{1}{1+e^{-s(x-c)}}, \tag{6.3}$$

where s is the slope and c is the centre. Note that while values $s = 1$ and $c = 0$ produce the standard sigmoid (Figure 6.2a), these parameters are trained (as seen in Section 6.1.2), to produce the intended scaling.

Genes from all diseases in M outside of the query q are collected in an intermediate weight vector X^q of size n. The weight X_i^q is the Caniza similarity between the q and the disease d to which the gene i is associated. If gene i is associated to more than one disease, the highest weight is used. Formally:

$$X_i^q = \begin{cases} \max\{H_{qd}\} & \text{if } M_{di} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{6.4}$$

The vector X^q is further scaled by the real number h between 0 and 1, which represents how important are seeds from similar diseases compared to seeds from the query disease (this parameter is also learned as seen in Section 6.1.2). Finally, the QWS is given as vector Y^q by setting the known associations of query q with a value of 1. Formally:

$$Y_i^q = \begin{cases} 1 & \text{if } M_{qi} = 1 \\ h \cdot \sigma_{s,c}(X_i^q) & \text{otherwise} \end{cases} \tag{6.5}$$

Notice that the superindex in Y^q is dropped in other sections as the QWS always refers to a particular query q.

Significance of the sigmoid

The sigmoid is intrinsically related to a two-class classification problem which arises when deciding whether a gene is a seed disease gene (class C_1) or not (class C_2) based on the similarity value. The Bayes theorem defines an identity of conditional probabilities:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{6.6}$$

Given that a similarity value x can only belong to one of the two classes C_1 or C_2, the law of total probability states that:

$$P(x) = P(x|C_1)P(C_1) + P(x|C_2)P(C_2). \tag{6.7}$$

Therefore, the posterior probability of a class given a similarity value can be written as:

$$P(C_i|x) = \frac{P(x|C_i)P(C_i)}{P(x|C_1)P(C_1)+P(x|C_2)P(C_2)}. \tag{6.8}$$

The equation can be simplified by calculating the log odds between the classes (the solution for C_1 is given without loss of generality):

$$r = ln\frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)}. \tag{6.9}$$

Equation 6.8, for $i = 1$, can be expressed in terms of r as:

$$P(C_1|x) = \frac{1}{1+exp(-r)} = \sigma(r), \tag{6.10}$$

which is the sigmoid function [14].

Therefore, by using the sigmoid function, the value assigned to each gene in the QWS for a query disease can be interpreted as the posterior probability for that gene to be a disease gene for the query.

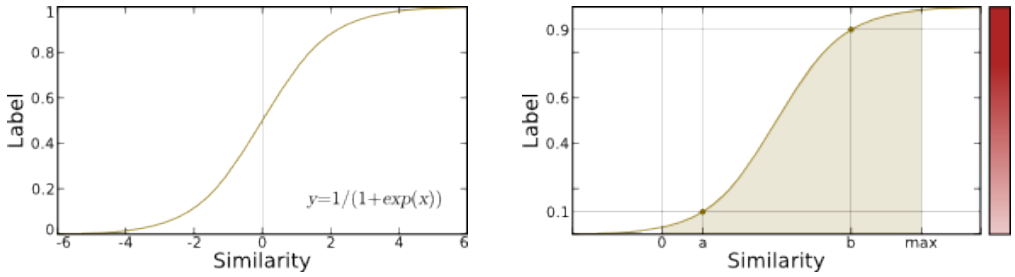
6.1.2 Training

Cardigan includes one hyperparameter for the diffusion process and three hyperparameters to produce the QWS. The first parameter is the diffusion regularisation parameter α from Equation 6.2. The following two parameters s and c are the sigmoid scale from Equation 6.3. The final parameter h is the regularisation parameter from Equation 6.5.

While the Caniza similarity can be scaled by modifying s and c directly, the relation between the similarity value and the expected result is not obvious within this formulation. Therefore, parameters s and c are obtained indirectly in such a way that a given (low) similarity value a is transformed to 0.1, and another (high) similarity value b is transformed to 0.9, as shown in Figure6.2b. The values s and c can be obtained for any pair of values $0 < a < b$, by solving an equation system of two variables:

$$\sigma_{s,c}(a) = 0.1$$
$$\sigma_{s,c}(b) = 0.9$$

where the $\sigma_{s,c}(b)$ is defined as in Equation 6.3. Note that the disease similarities between a and b belong to the pseudo-linear domain of the sigmoid.



(a) Standard sigmoid (b) Modified sigmoid

Fig. 6.2 **Visualisation of the standard and modified Sigmoid functions.** The Caniza similarity goes from 0 to a real *max* value. The sigmoid is used to convert the similarity to a range between 0 and 1. (a) Shows a regular sigmoid function for reference. (b) Shows a sigmoid function where the slope and centre are modified to associate disease similarity a to a low value (0.1) and similarity b to a high value (0.9). The colourbar on the side illustrates the amount of label gained after the transformation.

The default parameters are obtained by a greedy procedure to iteratively pick the parameter value that maximises Cardigan's performance at predicting genes for Acute Lymphoblastic Leukemia (MIM:613065) using 2013 data. This disease is chosen because it gained the largest amount of genes between the 2013 and 2017 editions of the OMIM database. The test evaluates the performance of Cardigan when all but one gene (among ETV6, RUNX1, PAX5, IKZF1, ABL1, FLT3, NOTCH1, and PI3K) are removed at a time, and are simultaneously predicted back –i.e. a *keep-one-in* experiment, where the performance is evaluated with the ROC AUC normalised for the first 200 false positives (see Section 4.2.1 for more details).

First, parameter α is tested between 0 and 1 in intervals of 0.01, since the parameter belongs to a bounded interval. Then, empirical observation of the Caniza matrix has shown that the similarity values range between 0 and 4 for both 2013 and 2017 versions. Therefore, parameters a and b are tested simultaneously between 0 and 5 ($0 \leq a < b \leq 5$) for intervals of 0.1. Notice that the similarity distribution is not linear, and the 99 percentile of similarity boundary lies roughly at the 2.5 value [20]. Finally, parameter h is also tested between 0 and 1 in intervals of 0.01. Notice that while a grid search over the parameters would be a more principled approach, this granularity would require $\sim$ 2300 days to finish instead of $\sim$ 6 hours, assuming 1 second per prediction (estimated from Table 6.1) and considering 8 predictions per configuration (one for each different seed gene). A simple coarse granularity grid analysis during the design revealed that the performance was most sensitive to the pair of parameters a and b, thus they are scanned together over a grid. The difference between several clusters of good configurations that appear with

the chosen greedy procedure seemed small enough to not require the usage of more sophisticated parameter selection techniques. However, an improvement for future work could be the usage of a Bayesian Optimization framework for the parameter selection.

The experiments use the following parameters: the binary networks HPRD, BioGRID and DiamondNet use $h = 0.50$, $a = 3.0$, $b = 4.0$, and $\alpha = 0.69$; while the weighted networks FUNCOUP and HIPPIE use $h = 0.60$, $a = 3.0$, $b = 4.0$, and $\alpha = 0.75$.

The decision to define default parameters follows the idea that Cardigan's usage should be as simple as possible (as the software implementation is delivered with the method). Furthermore, the exploratory analysis during the developmental stage showed that the parameters encountered few clusters of configurations close to good local maxima in performance, and those clusters were similar for several diseases. While the global maxima varied from disease to disease, the performance on the other local maxima appeared to be competitive. An in-depth analysis of Cardigan's performance with exhaustive parameter training appeared too expensive for the possible benefits. Notice that the parameter training requires to run Cardigan x amount of times per gene-association predicted in the train set (currently $x = 2700$ or ~ 45 minutes per gene-association). This would amount to ~ 160 days of training for all gene-associations of the 2017 OMIM (which would be required for a cross-validation analysis). Additionally, each fold of cross-validation would require $\sim$ 85 minutes for evaluation (which makes a leave-one-out cross-validation rather unfeasible). The aforementioned Bayesian Optimization framework could also make this analysis feasible in future work.

6.2 Results

Cardigan's performance is compared against ProDiGe1, ProDiGe4 and DIAMOnD at predicting disease genes. Following previous authors [135, 206, 231], the evaluation consists in predicting one gene at a time and measuring how often that gene is found within the first $1, 10, 100, 200$ genes output by the different algorithms. Further details can be seen in Section 4.2.1.

The evaluation of charted and uncharted diseases is presented separately, and for each type of disease the performance is analysed using both time-lapse data and a leave-one-out testing procedure. In time-lapse data experiments, the prediction attempts to retrieve genes which have been associated with diseases in the period 2013-17 using data from 2013. Although these experiments are limited in the size of

the test set, they are very important as they provide an evaluation of the system in real life scenarios. In leave-one-out experiments, a single disease-gene association is removed and the evaluation tells how well the system can retrieve it. Finally, as a baseline, the evaluation reports the performance obtained by a procedure that selects disease genes at random. The following results are presented for all the unweighted networks: HPRD, DiamondNet and BioGRID (see Section 4.3.2 for more details).

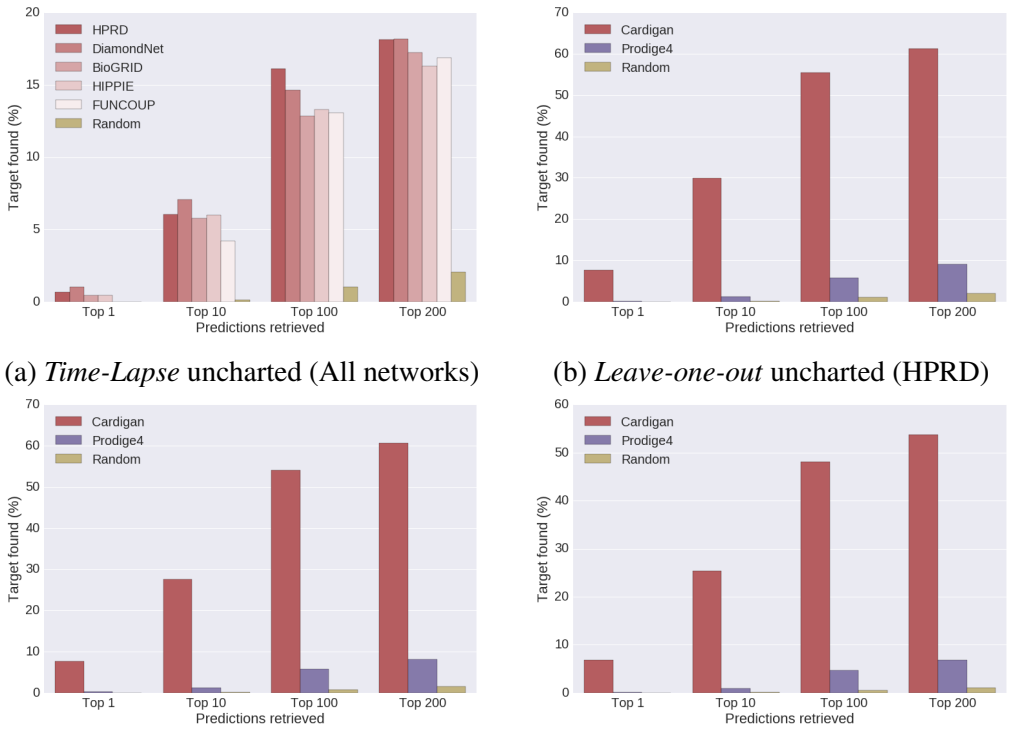
6.2.1 Performance on Uncharted diseases

Time-lapse tests

The first evaluation shows the performance of Cardigan at predicting genes which are associated with diseases in 2017 that were uncharted in 2013, using data from 2013. The 2013 OMIM database has 2670 descriptions of uncharted diseases, and 289 of those diseases appear as charted in the 2017 OMIM database. Cardigan is the only method that can make predictions for these 289 diseases. In fact ProDiGe4, the only other method that could in principle make predictions for uncharted diseases, is not applicable since its disease kernel does not include any of these diseases [205]. Figure 6.3a aggregates the prediction results of the 252 diseases which had genes found in at least one PPI network, and shows that Cardigan has a good performance which is stable across the different networks.

Leave-one-out tests

If a given disease has only one known disease gene, then removing it yields a synthetic uncharted disease. There are 5707 diseases with a single disease gene in the 2017 OMIM database, for 3,253 of them the disease gene was present in HPRD, 3,870 are included in DiamondNet and 4,152 are found in BioGRID. For each of these diseases its gene was removed and the performance of the methods at predicting it back was measured. Since these are synthetic uncharted diseases, there is no initial set of disease genes, and therefore ProDiGe4 and Cardigan are the only methods that can be used for this problem. Figures 6.3b, 6.3c and 6.3d show that Cardigan clearly outperforms ProDiGe4 for different numbers of retrieved predictions, and the difference is not an artefact of the network used.



(a) *Time-Lapse* uncharted (All networks)

(b) *Leave-one-out* uncharted (HPRD)

(c) *Leave-one-out* uncharted (DiamondNet)

(d) *Leave-one-out* uncharted (BioGRID)

Fig. 6.3 **Disease gene prediction for uncharted diseases.** Percentage of disease genes found in the predictions vs. the number of predictions retrieved. (a) Performance for diseases which were uncharted in 2013, but were charted in 2017, measured on different PPI networks. The percentage is normalised for the amount of genes available per network. (b), (c) and (d) Performances for a *leave-one-out* testing for diseases with a single known gene in 2017 on HPRD, DiamondNet and BioGRID respectively.

6.2.2 Performance on Charted diseases

Time-lapse tests

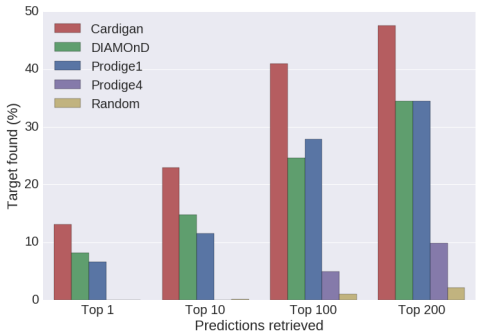
These experiments test the performance of the different methods at predicting genes for diseases which were already charted in 2013 and gained further genes by 2017, using data from 2013. Out of the 1413 disease gene associations which were new in the 2017 version of OMIM, only 95 of them were added to diseases which were already charted in 2013. This number is further reduced for testing since many of these genes were not contained in the PPI networks (their number ranges between 61 for HPRD, 71 for DiamondNet and BioGRID, and 78 for FUNCOUP). Results for HPRD are shown in Figure 6.4a, where Cardigan presents a minimum improvement of 38% with respect to the second best method at any threshold. On all networks DIAMOnD starts with a better performance than the remaining methods and ProDiGe1 becomes the second best by the top 100.

Leave-one-out tests

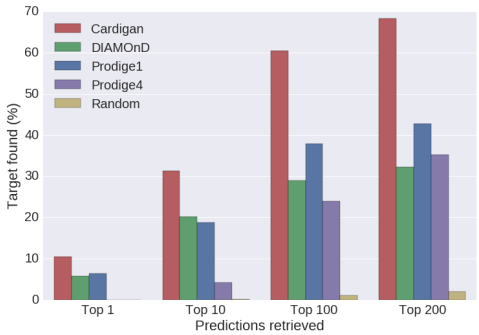
These experiments test the performance of the methods when disease genes known in 2017 were removed one at a time and predicted back. The 2017 OMIM database contains 264 diseases with two or more genes, which result in 970 possible test cases. Out of the 970 test cases, 769 tests can be performed using HPRD, 875 with DiamondNet and 893 with BioGRID. Figure 6.4b shows how Cardigan outperforms every method at every threshold – the minimum performance improvement is 55% with respect to the second best method at any given threshold. Results on the other networks are similar, while in DiamondNet (Figure 6.4d) the difference is closer between Cardigan and other methods, it is wider in BioGRID (Figure 6.4f). With single exception of DIAMOnD, that performs with the same accuracy on HPRD and DiamondNet over all thresholds, the methods show a reduced performance on the bigger networks.

6.2.3 Performance on Disease Module detection

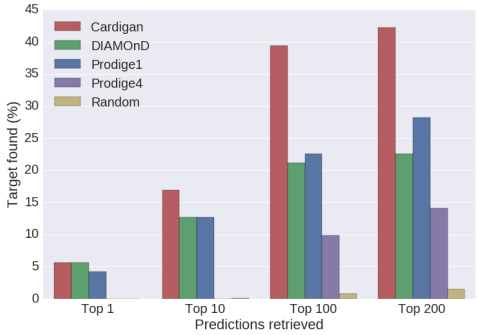
Finally, these experiments test how well Cardigan performed at predicting disease modules, i.e. whether the set of predicted disease genes formed a coherent disease module. These experiments follow the procedure and use the same dataset as presented previously by Ghiassian *et al.* [55]. Their dataset contains 70 diseases and their respective modules, which had been manually curated. The experiments evaluate the performance of Cardigan at reconstructing the module after removing different percentages of genes (i.e. keeping different percentages of the module). The



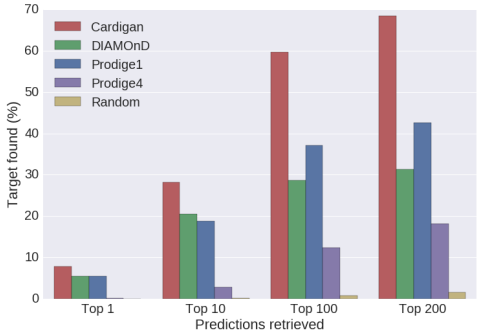
(a) *Time-lapse* charted (HPRD)



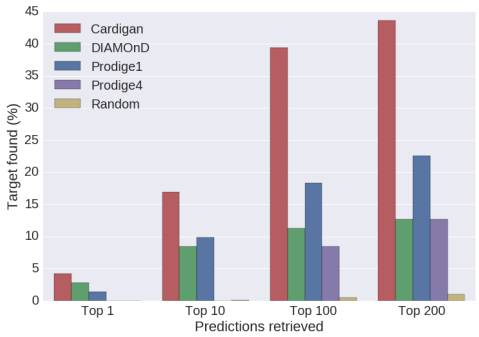
(b) *Leave-one-out* charted (HPRD)



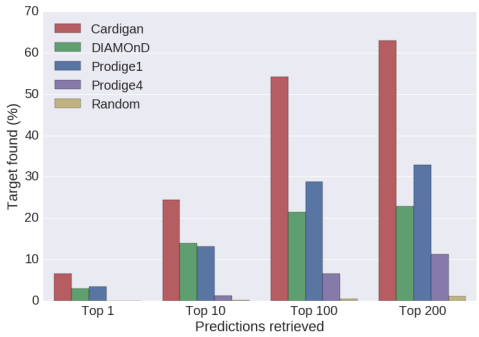
(c) *Time-lapse* charted (DiamondNet)



(d) *Leave-one-out* charted (DiamondNet)

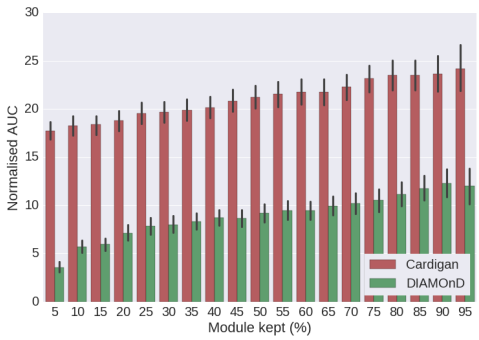


(e) *Time-lapse* charted (BioGRID)

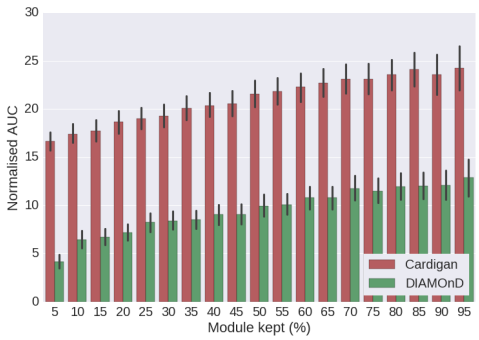


(f) *Leave-one-out* charted (BioGRID)

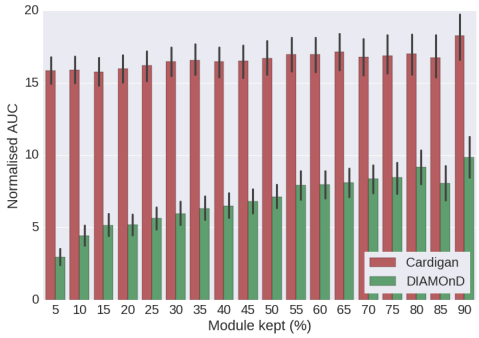
Fig. 6.4 **Disease gene prediction for charted diseases.** Percentage of disease genes found in the predictions vs. the number of predictions retrieved. (a),(c) and (e) Performance for predicting the genes that charted diseases have acquired between 2013 and 2017, on HPRD, DiamondNet and BioGRID respectively. (b),(d) and (f) Performances for a leave-one-out testing using 2017 data, on HPRD, DiamondNet and BioGRID respectively.



(a) HPRD



(b) DiamondNet



(c) BioGRID

Fig. 6.5 **Performance at reconstructing disease modules.** Different percentages of disease modules from Ghiassan *et al.* are removed and modules are then reconstructed. The *y*-axis shows the AUC of the ROC curve normalized for the first 200 false positives predictions. Error bars were calculated using the results for all diseases, each one with 10 random selections of kept genes. The expected value for a random prediction is 0.021 for HPRD, 0.0073 for DiamondNet and 0.0066 for BioGRID.

evaluation measure used is the AUC of the ROC curve normalized for the first 200 false positives predictions, thus matching the sizes of disease modules as described by Ghiassian *et al.* (for more details see Section 4.2.1).

Figure 6.5 shows that Cardigan outperforms DIAMOnD consistently when keeping different percentages of the module on all networks. At each percentage, we performed 10 random selections of the genes that were kept for each disease to avoid biases on the experiments. The minimum improvement is 85% when 80% of the module is kept on BioGRID. The performance between the methods decreases smoothly as the percentage of the module kept also decreases. However, DIAMOnD suffers a bigger loss when only 5% of the module is kept, where the difference in performance goes up to 450% when 5% on HPRD. Note how Cardigan is also able to recover modules even when 0% of the module is kept.

Table 6.1 **Average run times in seconds for a single prediction on different interactomes.** The averages were taken from the same test set with over 100 predictions on the same system. Intel XEON 2.6GHz, 32 GB RAM running Debian Jessie.

	Network				
	HPRD	*DiamondNet*	*BioGRID*	*HIPPIE*	*FUNCOUP*
Cardigan	0.48	0.62	0.92	0.75	1.10
DIAMOnD	1.32	3.86	7.73	-	-
ProDiGe1	2.59	4.79	9.98	-	-
ProDiGe4	176	291	433	-	-

6.2.4 Execution times

Although the execution times of the methods are not the main interest of this work, Cardigan is also significantly faster than DIAMOnD and ProDiGe. Table 6.1 shows the average prediction time for a prediction instance, which is the same for gene or module prediction – i.e. the retrieval of a single target for disease gene prediction or multiple targets for module prediction. These measurements do not include the initial time to load data (i.e. loading any matrices or parsing the interaction data to build the graph representations) into memory for any of the methods.

Notably, all the presented methods except DIAMOnD produce a ranking for all genes in the network, while DIAMOnD is only producing 200 predictions. As opposed to Cardigan and ProDiGe, iterative nature of DIAMOnD allows the procedure to stop when a certain number of results is produced.

6.3 Discussion

This chapter presents a novel network medicine based approach for disease gene prediction. Its key feature is its ability to predict genes for diseases using only their phenotypic description, which allows the method to predict genes for uncharted diseases. The approach can be thought of as establishing the location for the modules of charted diseases and using these to *triangulate* the location of the modules of uncharted diseases by exploiting disease phenotypic similarities.

The experiments show that Cardigan can handle networks of different sizes, for both weighted and binary edges, by testing it on HPRD, DiamondNet, BioGRID, HIPPIE and FUNCOUP. Furthermore, they demonstrate that Cardigan consistently

outperforms by a significant margin state-of-the-art methods and is stable on different types of networks. In particular, Cardigan's performance remains very high on BioGRID where other methods show significant drops in performance.

The difference in performance between Cardigan and the other methods is larger in time-lapse experiments than in leave-one-out tests, which are more commonly used in the literature. Notably, the former evaluation appears more significant than the latter, since time-lapse experiments provide a more realistic evaluation of disease gene prediction methods as they mimic more closely the gene discovery process. In fact, looking at the evolution of the OMIM database, genes for complex diseases are frequently discovered (and then added) in groups. The case of adding just one gene at a time, that is portrayed by *leave-one-out* tests, is much less frequent.

While the difference in execution times among the different methods might be negligible on a the research of a single disease, it resulted to be a hindrance for the experiments presented here. In particular, execution times of ProDiGe4 are within 2 and 3 orders of magnitude higher than Cardigan. Synthetic leave-one-out tests involve over 3,000 predictions; while Cardigan takes up to two hours for experiments on FUNCOUP, ProDiGe4 takes weeks to finish on HPRD the same hardware. This evaluation was performed on HPRD, DiamondNet and BioGRID.

Combining the results over all PPI networks from the time-lapse experiments and considering results only among the top 200 genes, Cardigan produces the best gene ranking for 80% of the diseases. Table 6.2 compiles some interesting examples of Cardigan predictions diseases using the 2013 OMIM database, which were later verified. It includes diseases which had been studied for long periods of time and yet, in 2013, were still missing associated genes – all these diseases have papers in OMIM dated at least from the '70s. Moreover, Cardigan was run on the entire OMIM diseases set (without removing any seeds) and the results are available at paccanarolab.org/cardigan. These constitute Cardigan's predictions for the missing genes for the real uncharted diseases, and additional genes for the charted diseases found in the 2017 OMIM database. While most charted diseases from OMIM are known to be monogenic and do not require additional genes to be discovered, the predictions can still be used to identify the disease modules. Most importantly, this table can be used as a starting point for the experimental discovery of disease genes for uncharted diseases.

Table 6.2 **Examples of Cardigan predictions using 2013 data.** All the presented diseases appeared in the 2013 OMIM database and had multiple papers associated with them, describing clinical features, inheritance or molecular genetics. However, OMIM did not include the associations with genes shown in the third column as they first appeared in the paper shown in the last column. The position on the Cardigan predicted ranking is also shown.

Disease	*2013 Status*	*Gene*	*Position*	*Paper*
Fetal Akinesia Deformation Sequence (MIM:208150)	Charted	MUSK	1	Tan-Sindhunata *et al.* (2015) [196]
Schimmelpenning-Feuerstein-Mims syndrome (MIM:163200)	Charted	NRAS	1	Lim *et al.* (2014) [109]
Familial Retinal Arteriolar Tortuosity (MIM:180000)	Uncharted	COL4A1	5	Zenteno *et al.* (2014) [232]
Ablepharon-macrostomia syndrome (MIM:200110)	Uncharted	TWIST2	10	Marchegiani *et al.* (2015) [120]

Chapter 7

Predicting genes from SNPs and Interfaces

This chapter explores the the usage of inferred protein interfaces (from the Interactome INSIDER database [130]) to build a binary PPI, and a modification for graph diffusion algorithms [236] to include the interfaces and the localisation of the SNPs. These modifications are designed to fit within the general approach defined for Cardigan, and when integrated yield an interesting prototype extension for Cardigan. The new network graph has nodes that represent protein interfaces, and edges that represent their interactions. The diffusion algorithm tackles the problem of diffusion for proteins with multiple interfaces (which appear disconnected in the graph). The algorithm also identifies of particular interfaces where disease SNPs are located to prioritise the diffusion on a particular area in the graph. This information is fed into the model through the initial labelling vector.

The experimental characterisation of protein interfaces entails the analysis of co-crystal protein structures, which usually banks on the expensive X-ray crystallography [189], NMR, and more recently cryo-EM [98]. These structural approaches identify residue interactions from the quaternary compound structure [97, 157] by localising close residues from the two interacting proteins (e.g. a radius of 4 Å), and other structural properties [93]. While computational methods are known to be successful in the identification of protein interfaces, structural modelling approaches have extremely high CPU cost [93].

Another trend followed by computational methods, that avoids the construction of these quaternary structures, is the evaluation of biophysical properties to predict interfaces on particular study cases [108, 159, 38, 81]. A recent computational framework by Meyer *et al.*, called ECLAIR (Ensemble Classifier Learning Algorithm to predict Interface Residues), predicts the interfaces on a full interactome,

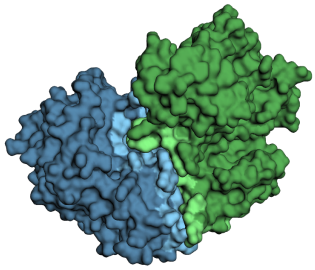


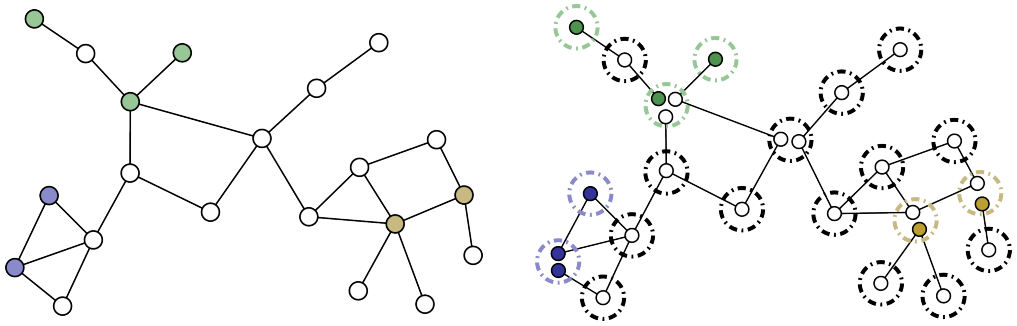
Fig. 7.1 **Example of a co-crystal structure and the interaction interfaces.** The light regions in the picture represent the residues which lie within the interaction interface. Notice that while these residues are close in the quaternary and tertiary structures (i.e. 3D space), they may be distant in the primary structure (i.e. 1D sequence) of the protein. Picture from Meyer *et al.*[130].

while improving on the performance of other feature based methods [130]. ECLAIR models each residue on a protein pair through a series of features, ranging from biophysical properties, evolutionary [114, 134], structural features [156], and docking predictors [71, 157]. Multiple aggregation strategies are used to produce variations of the features, and which serve to train a collection of classifiers. An ensemble of the outputs of 8 independent classifiers is used as the overall ECLAIR prediction.

This proposal is certainly novel, as experimental protein interaction interfaces are scarce, and computational predictions for an interactome level of coverage are recently available from INSIDER, a database which aggregates high quality ECLAIR predictions [130]. The extension can be considered to provide Cardigan the capacity to use networks with interfaces, in addition to binary and weighted PPIs. The usage of a network with interfaces appears to be interesting for network methods, as its usage allowed to enhance the performance of Cardigan on Charted diseases and disease modules.

7.1 Predicting in a protein-interface network

The proposal starts from a data prepossessing step to model the interfaces of each protein from the interface residues contained in INSIDER. Ideally, the interfaces of a protein are such that it could simultaneously interact to two other proteins if they bind on different interfaces, but cannot simultaneously interact with proteins that bind on the same interface. These protein interfaces are the nodes in the new graph, and the edges represent the interacting protein interfaces (see Figure 7.2). Following the Cardigan approach, the production of a protein interface Query Weight Set is shown in Section 6.1.1. In order to include information about disease gene variants



(a) (b)

Fig. 7.2 **Illustration of a protein interface graph.** Genetic variants of different diseases (blue, green and yellow) are highlighted on the network. (a) A protein interaction network. The solid circles represent proteins which are the nodes of the graph, and the edges represent protein-protein interactions. (b) A protein interface interaction network. The dashed lines represent proteins, and the solid circles represent the interfaces, which are the nodes of the graph, and edges are interactions between particular interfaces in a protein.

and SNPs, the known associations are mined from ClinVar [102] instead of OMIM (more details about ClinVar are shown in Section 4.3.1).

Notice there may be some topological differences due to modelling a PPI network as a protein interface graph instead of a protein graph. Changes on the number of connected components, average node degree, and other graph properties can be expected since the number of edges is likely to remain constant, while the number of nodes increases. This is illustrated in Figure 7.2b, where the protein graph (left) is connected, while the protein interface graph (right) has five different connected components. Ideally, the differences in the topology serve as a better representation of the interactome and aid the diffusion process. For instance, the yellow disease appears to belong to a big cluster in the right of Figure 7.2a, however Figure 7.2b suggests the existence of two different modules. In particular the disease related module appears to bottom of the cluster, where the two interfaces with a variant are located. Table 4.4 quantifies the main differences of the graphs used in this analysis.

The structure of the protein interface graph may also produce some unwanted effects. If the different interfaces of a single protein are completely independent elements, interactors on different interfaces (of the *middle* protein) are considered infinitely far apart (as opposed to 2 hops away in a protein graph). This effect could limit the ability of the diffusion in considering compounds associated to a disease, as the diffusion would not carry the label through the different interfaces of the *middle* protein. The problem is illustrated by the green disease from Figure 7.2,

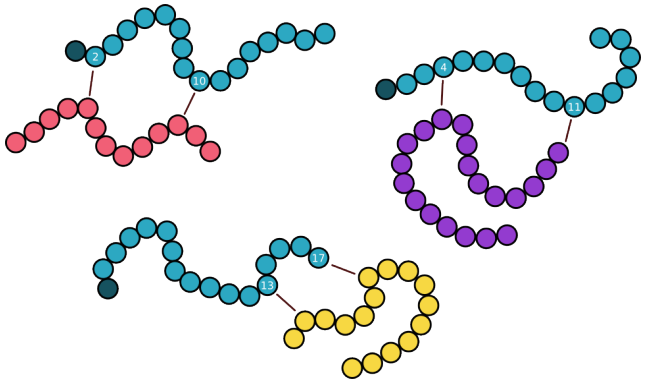


Fig. 7.3 **Sketch of overlapping protein interfaces.** The sketch depicts the primary structure of interacting proteins (cyan, which interacts with pink, purple and yellow), where the interacting residues are connected with straight lines. The first residue is highlighted to orient the protein. The cyan protein interacts with pink through residues 2 and 10, with purple though 4 and 11, and with yellow though 13 and 17. The proposed procedure defines two interfaces for the cyan protein: one from residue 2 to 11, and another from 13 to 17. This considers that the pink and the purple proteins are not likely to be able to dock simultaneously.

where all genes appear close in the protein graph (left), and spread over two different connected components in the protein interface graph (right).

Two variants of the diffusion procedure are presented to account for the fact that the different interfaces of a protein should not be treated as completely independent: 1) The first variant transforms the protein interface graph to a protein graph by a a linear combination between the nodes corresponding to the different interfaces of each protein. Thereafter it follows the steps defined by the Cardigan pipeline from the semi-supervised Zhou *et al.* [236] diffusion. 2) The second variant presents a novel formulation of a diffusion process with regularisation on an interface graph. This yields an interface ranking instead of a protein ranking as an output. Therefore, the final protein ranking is produced from the first observation of any given interface of the protein.

7.1.1 Building the interaction network

While INSIDER includes a few gold standard interactions with experimental validation, most are high quality ECLAIR predictions [130]. Each INSIDER interaction consists in two sets of residues: the interface residues of protein A, and the interface residues of protein B. As low quality ECLAIR predictions are discarded, an INSIDER interaction entry may be missing the set of residues for one of the proteins.

The INSIDER database can be directly transformed to a protein residue interaction graph $G = \langle V, E \rangle$, where the nodes $V = \langle p, r \rangle$ are tuples of proteins p and

interface residues $r = \{r_i\}$, that are given as a set (which might be empty $r = \varnothing$); and the edges are:

$$(\langle p, r\rangle, \langle q, s\rangle) \in E,$$

where r and s are the interacting residues of proteins p and q respectively. Notice that each residue may be part of several interfaces, and two interfaces may be different even if they contain a common residue.

The following is a naive procedure to combine pairs of residue interfaces into *interaction hotspots*, where different interactor proteins are expected collide if they were to bind simultaneously (see Figure 7.3). The procedure only considers the protein residues within hotspots to be relevant in the construction of the interface graph, so there is no need for every protein residue to belong to a particular hotspot.

First, the set of residue interfaces r of each node $\langle p, r\rangle$ is extended as $\bar{r}$, to include all residues in between the minimum and maximum element of the set, which yields the extended tuples $\langle p, \bar{r}\rangle$. The naive intuition is that the interacting residues and all residues between them are *blocked* during the protein binding. All *blocked* residues will belong to the interaction hotspot. Formally, each tuple $\langle p, r\rangle$ is transformed into an extended tuple $\langle p, \bar{r}\rangle$, such that:

$$\min\{r\} \leq r_i \leq \max\{r\}. \tag{7.1}$$

Then, all the extended tuples $\langle p, \bar{r}\rangle$ with overlapping residues $\bar{r}$ are combined until the remaining sets are disjoint, producing the interaction hotspots[1] $(p, \mathbf{h})$. The naive intuition is that a pair of interactors will collide if they *block* the same residue[2]. Formally, the interaction hotspots $(p, \mathbf{h})$ of a protein p are such that:

$$\begin{array}{rcll} \bar{r} \cap \bar{s} \neq \varnothing & \rightarrow & \bar{r} \cup \bar{s} \subseteq \mathbf{h} & \exists (p,\mathbf{h}) \forall \langle p, \bar{r}\rangle, \langle p, \bar{s}\rangle \quad \wedge \\ \mathbf{h} \cap \mathbf{k} & = & \varnothing & \forall (p,\mathbf{h}) \neq (p,\mathbf{k}) \quad \wedge \\ \bigcup\limits_{\forall \langle p, \bar{r}\rangle} \bar{r} & = & \bigcup\limits_{\forall (p,\mathbf{h})} \mathbf{h} & \end{array} .$$

The procedure ends with the graph $G_\phi = \langle V_\phi, E_\phi \rangle$, where the vertices and edges are translated to protein hotspots. The vertices are simply the set of all protein

[1] Parenthesis are used instead of angle brackets to add clarity in the notation.

[2] The naive procedure clearly does not account for several 3-dimensional properties of proteins. For instance, it does not consider that a residue chain may fold and expose the ends of a residue chain in one area of the surface, while the middle of the chain is exposed in another; nor does it consider to block residues beyond the known interface residues, which may also be obstructed in the 3-dimensional space by an interactor. These (and other) considerations could be explored in further research.

hotspots $V_\phi = \{(p, \mathbf{h})\}$. The edges E_ϕ are such that if two interfaces interact, then, the hotspots containing those interfaces interact. Formally:

$$(\langle p, r\rangle, \langle q, s\rangle) \in E \rightarrow \{((p, \mathbf{h}), (q, \mathbf{k})) \in E_\phi \,|\, r \subseteq \mathbf{h} \wedge s \subseteq \mathbf{k}\}.$$

Notice that with this procedure, the empty interfaces $\langle p, \varnothing\rangle$ are not combined into other hotspots, and yield equivalent elements $(p, \varnothing)$. This is an intended feature because they can be overlapping with any of the defined interfaces, or possibly through a different still unknown interface.

While the hotspots are not precisely protein interfaces, they represent the same concept for all intended uses of this network. Henceforth the binary undirected graph $G_\phi = \langle V_\phi, E_\phi \rangle$ is referred as the interface network, with edges E_ϕ between protein interfaces in V_ϕ.

For a future simplification of notation the set of all the interfaces of a protein p is represented as $\{p, *\}$. The set of edges incident on an interface $\mathbf{h}$ is denoted as $\{(p, \mathbf{h}), *\}$, and the edges incident on a protein p as $\{\{p, *\}, *\}$.

Following this procedure, the INSIDER database yields the interactome with interfaces, which is referred to as the INSIDER ECLAIR network (represented by G_ϕ). However, INSIDER can be used to produce a simple interactome by connecting pairs of proteins which have interacting interfaces (essentially ignoring the interfaces), which is referred to as the INSIDER PPI (compared in 4.4).

7.1.2 Building the Query Weight Set with interfaces

Analogously to Cardigan, the *Query Weight Set* (QWS) is a collection of weights associated to the protein *interfaces* all known diseases, and it is calculated for each query disease given. The disease-gene associations are mined from ClinVar [102], which details the particular SNPs associated to each disease. The construction presented here tweaks the procedure from Section 6.1 to associate the weights to particular protein interfaces when possible.

The QWS is produced by an iterative procedure, based on the Caniza *et al.* similarity [20]. On each iteration, the unweighted proteins associated with the disease most similar to the query not yet visited are collected. Then, the similarity between the disease and the query is scaled by a sigmoid function. If the SNP occurs in a particular interface, it is associated with the weight; otherwise the weight is

uniformly distributed to all the protein interfaces[3]. The process continues until all diseases in the ClinVar database are visited.

Other databases that associate gene variants to phenotypes (such as HGMD [193] or DisGeNET [158]) could be used instead of ClinVar, although it is possible that the method parameters would require to be trained again.

7.1.3 Variant 1: Compacting the network

This variant is explained easily following the matrix representation of the consistency method by Zhou *et al.*. The procedure entails calculating the matrix representation of the interface problem (ϕ), and a linear combination of the elements corresponding to the same protein reducing to a protein matrix representation (ρ). Recall that the label propagation procedure diffuses the initial labels from the QWS (stored in Y) in order to minimise the cost function $C(F)$ from Equation 6.1. This has the matrix representation from Equation 6.2:

$$F = \beta(I - \alpha S)^{-1}Y,$$

where $K = \beta(I - \alpha S)^{-1}$ is a diffusion kernel [236].

Let Y_ϕ be the initial label distribution given by the interface QWS (following Section 7.1.2), and K_ϕ the Zhou diffusion kernel derived from the interface graph $G_\phi = \langle V_\phi, E_\phi \rangle$. The elements of interface vector Y_ϕ are combined to form protein vector Y_ρ following:

$$Y_\rho[p] = \sum_{\mathbf{h} \in \{p,*\}} \omega[p,\mathbf{h}] Y_\phi[(p,\mathbf{h})] \quad \forall p \in V\rho,$$

where $Y[i]$ is element i of the initial vector, and $\omega[p,\mathbf{h}]$ is the fraction of interactors of protein p that bind on interface $\mathbf{h}$:

$$\omega[p,\mathbf{h}] = \frac{|\{(p,\mathbf{h}),*\}| + \delta[p]}{|\{\{p,*\},*\}|}, \quad \mathbf{h} \neq \varnothing$$

where $\delta[p]$ is the chance of getting extra links due to a void interface, defined as:

$$\delta[p] = \frac{|\{(p,\varnothing),*\}|}{|\{p,*\}|}.$$

[3]Recall that the current methodology is proposed for disease gene prediction, and is not intended to identify which interface is associated to the disease. Distributing the weight to all interfaces equates to flagging the entire protein as a seed for diffusion, as no interface can be identified as a preferential focus for the diffusion. Furthermore, SNPs might cause alterations to the protein folding or non-functional protein products, so it is sound to consider that a deleterious SNP may affect all interfaces.

To allow consistency between the amount of edges:

$$\omega[p,\varnothing] = \frac{\delta[p]}{|\{\{p,*\},*\}|}.$$

An analogous process is followed to reduce the dimensionality of S_ϕ and derive S_ρ as:

$$S_\rho[p,q] = \sum_{\mathbf{h}\in\{p,*\}} \omega[p,\mathbf{h}] \sum_{\mathbf{k}\in\{q,*\}} \omega[q,\mathbf{k}] S_\phi[(p,\mathbf{h}),(q,\mathbf{k})] \quad \forall p,q \in V_\rho,$$

where $S[i,j]$ represents row i and column j of matrix S. The protein-wise diffusion kernel is calculated as $M_\rho = \beta(I - \alpha S_\rho)$. This variant effectively transforms the interface network and corresponding diffusion to a protein network and provides a diffusion kernel to handle the nuance characteristics.

This model has proved to work empirically, however, the diffusion kernel does not match the mathematical formulation from Equation 6.1 after the transformation. This has the effect that there are no guaranties that the diffusion process will hold for any interface graph. The following variant proposes a formulation for a diffusion process for a graph with interfaces to address the lack of mathematical guarantees.

7.1.4 Variant 2: Diffusion on the interfaces

The second variant presents a full formulation of the cost function $C(F)$ for a interface graph $G_\phi = \langle V_\phi, E_\phi \rangle$, expanding on theoretical concerns of the Zhou *et al.* formulation. The cost function $C(F) = \frac{1}{2}(C_1 + C_2 + C_3)$ is composed of three terms, where the first penalises the difference in label between connected nodes (as given in the consistency method):

$$C_1 = \sum_{((p,\mathbf{h}),(q,\mathbf{k}))\in E_\phi} W[(p,\mathbf{h}),(q,\mathbf{k})] \left[\frac{F[(p,\mathbf{h})]}{\sqrt{D_W[(p,\mathbf{h}),(p,\mathbf{h})]}} - \frac{F[(q,\mathbf{k})]}{\sqrt{D_W[(q,\mathbf{k}),(q,\mathbf{k})]}} \right]^2 \tag{7.2}$$

where W is the weight matrix derived from the interface graph, and the diagonal degree matrix D for a generic matrix A has elements:

$$D_A[(p,\mathbf{h}),(p,\mathbf{h})] = \sum_{(q,\mathbf{k})\in V\phi} A[(p,\mathbf{h}),(q,\mathbf{k})].$$

The next addresses the concern of sharing the label across interfaces of the same protein. This term penalises the difference in label across all different interfaces of a protein:

$$C_2 = \lambda \sum_{p \in V_\rho} \frac{1}{|\{p,*\}|} \sum_{\mathbf{h},\mathbf{k} \in \{p,*\}} [F[(p,\mathbf{h})] - F[(p,\mathbf{k})]]^2, \tag{7.3}$$

where $\lambda \in (0,\infty)$ is the relative importance of the term, V_ρ is the set of all proteins in the interactome, and $|\{p,*\}|$ is the number of interfaces for protein p.

The final term is set to conserve the original labelling, where lower initial labels are given a lower penalisation if they change. This considers that a low seed represents that initially there is little information to associate that gene to the disease; it does not represent that there is information to keep the gene far from the disease. This is a crucial difference with similar formulations [136, 206, 236] where initial labels equal to zero are taken as true negative associations. Formally:

$$C_3 = \mu \sum_{(p,\mathbf{h}) \in V_\phi} G(Y[(p,\mathbf{h})]) \left(F[(p,\mathbf{h})] - Y[(p,\mathbf{h})]\right)^2, \tag{7.4}$$

where $\mu \in (0,\infty)$ is the relative importance of the term, the function $G(Y[(p,\mathbf{h})]) = \gamma + Y[(p,\mathbf{h})]$ establishes the importance of each initial label, and $Y \in [0,1]$ is the QWS for the protein interface graph. Here, $F_i - Y_i$ is the change in label and $Y_i + \gamma$ is the penalisation coefficient, where γ is a small positive value to represent the likelihood of the zero values of the initial labelling to be correct[4]. A lower penalisation coefficient equates to give further priority to the adjacent labels, so that node will approximate better the value of its neighbours when Y_i is low.

The addition of Equations 7.2, 7.3 and 7.4 yields $C(F)$. The following formulation uses a subscript notation instead of the full protein notation to ease the reading:

[4] A penalization coefficient of zero for all initial values equal to 0 could cause the diffusion method not to prioritise the module around the high seed values, breaking one of the core principles behind the methods presented in this work. Consider an extreme case with a single label equal to 1 and the rest equal to 0. That diffusion would be dominated by the labels of adjacent nodes and other interfaces (C_1 and C_2). A final labelling where all nodes of are equal to 1 would minimise that cost function since there are no differences between pairs of nodes. However, this would not prioritise the topological module around the seed.

$$C(F) = \frac{1}{2}\left[\sum_{i,j=1}^{|V_\phi|} W_{ij}\left\|\frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}}\right\|_2^2 + \lambda \sum_{i,j=1}^{|V_\phi|} B_{ij}\left\|F_i - F_j\right\|_2^2 + \mu \sum_{i=1}^{|V_\phi|} G_{ii}\left\|F_i - Y_i\right\|_2^2\right] \tag{7.5}$$

where the block diagonal matrix $B = diag(B_1, \ldots, B_n)$ with square blocks B_p of size $|\{p,*\}|$ connects the interfaces of a each protein p:

$$B_p[i,j] = \frac{1}{|\{p,*\}|} \quad \forall i,j \tag{7.6}$$

and the diagonal matrix $G = diag(Y_1 + \gamma, \ldots, Y_{|V_\phi|} + \gamma)$ represents the trust of each initial label Y. Notice that the main diagonal of G is strictly positive and therefore it is symmetric positive definite (SPD) as it is a diagonal matrix [27].

The matrix form of Equation 7.5 can be seen as:

$$\begin{aligned} 2C(F) &= F^T(I - D_W^{-1/2} W D_W^{-1/2})F + \lambda F^T(D_B - B)F + \mu(F-Y)^T G(F-Y) \\ &= F^T L_W^{sym} F + \lambda F^T L_B F + \mu(F-Y)^T G(F-Y) \end{aligned} \tag{7.7}$$

where L_W^{sym} is the symmetric normalized Laplacian of W, and L_B is the Laplacian of B [27].

Then, the optimum labelling F can be found equating the gradient of C (Equation 7.7) to zero ($\frac{\partial C}{\partial F} = 0$). This yields:

$$\begin{aligned} \frac{\partial C}{\partial F} &= L_W^{sym} F + \lambda L_B F + \mu G(F-Y) \\ (L_W^{sym} + \lambda L_B + \mu G)F - \mu G Y &= 0, \end{aligned} \tag{7.8}$$

since L_W, L_B and G are SPD, their sum is SPD (and is invertible) [27]. Therefore, F has the following closed form solution:

$$\begin{aligned} (L_W^{sym} + \lambda L_B + \mu G)F &= \mu G Y \\ F &= (L_W^{sym} + \lambda L_B + \mu G)^{-1} \mu G Y \end{aligned}. \tag{7.9}$$

Notice that GY produces a column vector and that $(L_W^{sym} + \lambda L_B + \mu G)$ is a SPD matrix. Therefore, the solution of the linear system can be found using the Conjugate

Gradient method. Recall that G is a function of Y, so the Equation matrix must be calculated for each prediction.

From this formulation the approach follows the Cardigan pipeline to produce the final results.

7.2 Evaluation

The performance of both Variants is compared directly to Cardigan on the prediction of ClinVar disease associations[5]. Other methods (i.e. ProDige1, ProDiGe4 and DIAMOnD) are excluded due to their subpar performance (see Section 6.2). The evaluation is centred on charted diseases as the purpose of the extension is to analyse the applicability of known genetic variants. Cardigan is run on the INSIDER PPI network, while Variant 1 and Variant 2 are run on the INSIDER ECLAIR network. This choice is made since the some topological components in the protein network can become disconnected when considering the interface graph (illustrated in the bottom-right cluster from Figure 7.2), which could affect Cardigan's performance.

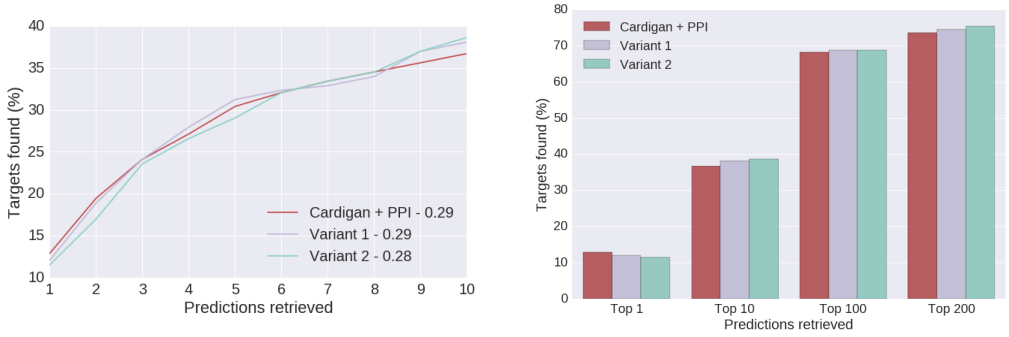
This configuration causes Cardigan and Variant 1 to directly rank INSIDER proteins, and Variant 2 to rank INSIDER protein interfaces. The current evaluation continues to be based on the identification of disease genes. Therefore a protein interface ranking is made comparable by ranking only proteins based on their first occurrence in the prediction. The identification disease interfaces, or regions in the disease protein which might contain SNPs is a completely separate problem. This problem will require an additional state-of-the-art review to build a new evaluation pipeline, which escapes the scope of this exploratory analysis.

The following experiments are exclusively composed of test sets from ClinVar diseases where the target genes are synthetically removed, and later predicted by the procedures. Gene predictions are evaluated using the recall at thresholds, and module predictions using the normalised ROC (details on Section 4.2.1). Only the latest version of the ClinVar database is found available [102], therefore *time-lapse* experiments cannot be performed (see Section 4.1).

Variant 1 solves $F = (I - \alpha S_\rho)^{-1} Y_\rho$ using all the default diffusion parameters from Cardigan.

Variant 2 contains a different formulation so its parameters were trained following Section 6.1.2. The procedure derived in $\lambda = 2.5$, $\mu = 3.8$, $h = 0.8$, $a = 2.0$ and $b = 3.0$. Additionally, the confidence in initial zeros was set as $\gamma = 0.01$.

[5]The ClinVar database provides an OMIM identifier for its diseases which allows the creation of a QWS (independently for Cardigan and the Variants) using the Caniza similarity. The QWSs created in these experiments contain only ClinVar diseases and gene associations.



(a) (b)

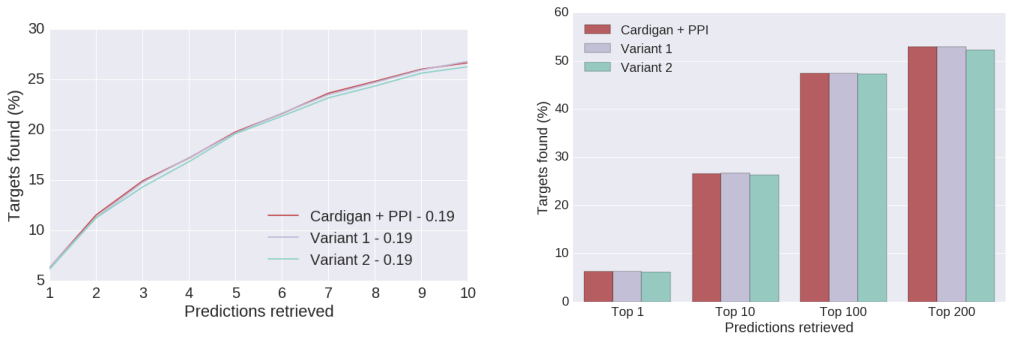
Fig. 7.4 **Testing Cardigan variants for charted diseases.** Performances for a *leave-one-out* testing for diseases with a single associated protein in ClinVar. Cardigan uses the INSIDER PPI, while the variants use the INSIDER ECLAIR networks. (a) Normalised ROC for the top 10 predictions. (b) Shows the fraction of disease genes found within the top 1, 10, 100, 200 predictions.

Charted diseases

The initial comparison is a *leave-one-out* charted experiment. It tests the performance of Cardigan and Variants when disease genes are removed one at a time and predicted back. There are 133 diseases with two or more associated proteins in ClinVar, which yield a total of 416 possible test cases. Out of the 416 cases, 365 can be performed using proteins from the INSIDER database. Figure 7.4a presents a normalised ROC curve for the first 10 predictions and shows how both Variants edge over Cardigan after a few elements. Figure 7.4b shows that the difference is 2% at 10 predictions, and is sustained up to 200 predictions. Notice that Variant 2 stars with the lowest performance, but it performs the best at 200.

Uncharted diseases

The second comparison is a *leave-one-out* uncharted experiment. If a given disease has only one known disease gene, then removing it yields a synthetic uncharted disease. There are 3474 diseases with a single gene association in ClinVar. Out of the 3474 possible test cases, 2967 can be performed using proteins from the INSIDER database. Figure 7.5 shows that Cardigan and the Variants perform very similarly when no genes are known. The trend where Variant 2 drops a bit in performance and catches up with the other methods seen among the top 10 results (Figure 7.5a), is consistent up to 200 predictions (Figure 7.5b). However, the difference is less than 1% and Variant 2 can be considered to produce high-quality results comparable to Cardigan.



(a) (b)

Fig. 7.5 **Testing Cardigan variants for uncharted diseases.** Performances for a *leave-one-out* testing for diseases with a single associated protein in ClinVar. Cardigan uses the INSIDER PPI, while the variants use the INSIDER ECLAIR networks. (a) Normalised ROC for the top 10 predictions. (b) Shows the fraction of disease genes found within the top 1, 10, 100, 200 predictions.

Disease modules

Finally, both Variants are evaluated as disease module prediction methods. For this prediction, the Ghiassian *et al.* disease module dataset [55] (see Section 4.3.1), as used for the Cardigan evaluations (see Section 6.2.3). The test consists in removing different percentages of known genes, and predicting all the targets simultaneously.

The Ghiassian dataset contains 70 diseases and the molecular basis is annotated with Entrez identifiers [117], which need to be translated for their usage with the INSIDER network. The proteins are mapped to UniProt [31] using the translation database provided by the HUGO gene nomenclature committee [226], rendering 1529 proteins and 2831 disease-protein associations available testing (99.6% translation success).

Figure 7.6 shows how both variants outperform Cardigan in different thresholds. Variant 1 performs better when the number of missing genes is lower, and Variant 2 performs better when large percentages of the modules are missing. Notice that the difference in performance shown is in terms of the Normalised AUC, and that many diseases produced no results, lowering perceived difference.

7.3 Discussion

Pros and cons of the Variants

The novel methodology allows incorporation of clinical variants and protein interfaces into a protein interface graph for a network medicine application in disease

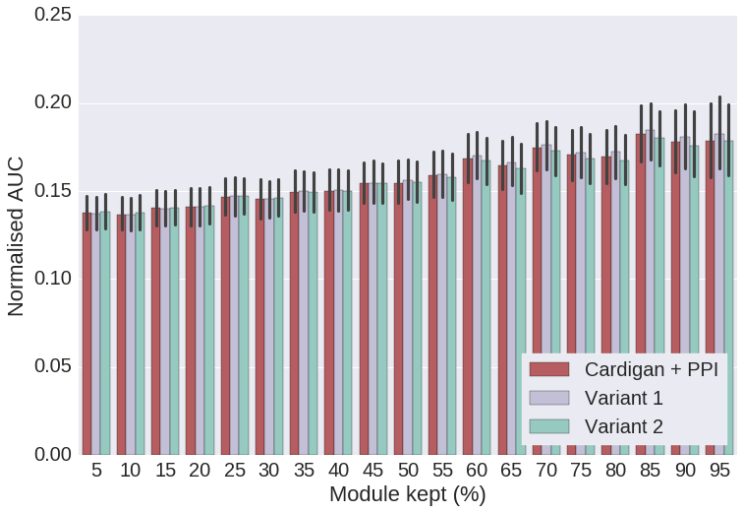


Fig. 7.6 **Performance at reconstructing disease modules.** Different percentages of disease modules from Ghiassan *et al.* are removed and modules are then reconstructed. The *y*-axis shows the AUC of the ROC curve normalized for the first 200 false positives predictions. Error bars were calculated using the results for all diseases, each one with 10 random selections of kept genes. The expected value for a random prediction is 0.007.

gene prediction. This can be considered an extension to Cardigan, which already contains interesting features such as allowing predictions for uncharted diseases and the inclusion of up-to-date phenotypic information.

In principle, different gene variants will affect different modules around the gene. The intuition is that the usage of interfaces allows the method to differentiates between the modules. This can be seen when the method prioritises the selection of candidates along the disrupted interface, while reducing the diffusion through the unaltered interfaces. The candidates along the disrupted interface can be directly affected by alterations in the interaction due to a mutant interface. Interactions through the unaltered interfaces should not be entirely ignored in the procedure as proteins found through those interfaces do interact with a disease protein. A complex formed with a disease protein will a potential effect on some biological process.

Variant 1 is produced to test the viability of using networks with interfaces, and it is remarkable by outperforming Cardigan on charted diseases. These results are encouraging to consider further development on the interface approach, with possible improvements on the interface combination procedure. Furthermore, the method can be extended to include other networks by a natural continuation of this Variant since $\varnothing$ interfaces are basically considered to be the normal PPI nodes, as opposed to the interface nodes. Additional networks can be trivially added to the graph using the same considerations given to the $\varnothing$ interfaces.

Unfortunately, Variant 1 presents a notable shortcoming regarding its formulation. The current incarnation essentially transforms a rich network with interfaces into a weighted protein protein network, through a heuristic procedure. While the extra information provided by the interfaces is encoded into these weights, the formulation of the diffusion procedure may lack mathematical rigour when handling the interfaces. It is therefore fundamental to produce a new mathematical formulation which considers the nature of this peculiar graph.

This shortcoming is addressed by Variant 2, which provides a cohesive framework that integrates the variables at play in this diffusion problem. Section 7.1.4 describes a theoretically sound diffusion process which includes a network with interfaces and a formal demonstration of the feasibility of the solution. The inclusion of known SNPs is natural in this formulation, as they are encoded in the QWS. The QWS is represented by vector Y, which not only determines the initial labels to diffuse, but transforms the underlying diffusion pattern as matrix μG (from Equation 7.9) is built with Y.

Variant 2 is not designed to improve the performance of Variant 1, nonetheless, the performance of Variant 2 is competitive in all the experiments (it is even the top performing method for some thresholds in Figure 7.4). This exceeds the expectations of the approach, and justifies the research of additional considerations to warrant a better overall performance.

The diffusion formulation from Variant 2 presents some notable characteristics, which could be usable in other diffusion methods. 1) The formulation shows how an additional term in the cost function can be used to share a label between disconnected nodes. Notice that the second term of the formulation (Equation 7.3) controls the similarity between sets of nodes that have no edges but are joined by a conceptual grouping (interfaces of the same protein are joined together). An analogous approach can be used to combine different networks which have the same nodes, but where the edges have different meanings. The idea is that the nodes of the independent networks can be seen as interfaces of the node in the aggregation network. This idea could be used to incorporate data such as protein complexes [140], Pfams [13], metabolic and transcriptomic data [3, 54] into a single graph suitable for a diffusion process. 2) The formulation subscribes to the positive-unlabelled (PU) learning paradigm, in which in the absence of interactions do not represent negative data. This consideration follows from the fact that experimental biological data is hard to obtain, and experiments rarely prove negative interactions or associations. Several methods use PU learning with diffusion [136, 206, 236]. However, this formulation adds further richness to the model as it considers that the low weighted labels are

more likely to be wrong than the high weighted labels. The application is seen in the third term cost term involving the initial labelling vector Y in Equation 7.4.

A drawback of this formulation is the inclusion of extra hyperparameters in the algorithm. The third term in Equation 7.5 adds the regularization parameter λ, and increases the training cost of the procedure. However, the hyperparameter training is not a time constrained step, since it is performed previous to all particular predictions. The time cost of an individual prediction of Variant 2 is comparable to a prediction from Cardigan and takes less than a second.

Prediction of disease gene or drug binding interfaces

The prediction of *disease gene interfaces* is a new problem that appears tractable with the current proposal (Variant 2). However, this might prove a bit tricky given the low amount of data current available. Notice the nodes representing $\varnothing$ interfaces do not identify particular regions in the protein, so they do not serve to distinguish between different interfaces in a protein. Table 4.5 shows that only about 2% of the proteins have 2 or more predicted interfaces (discarding $\varnothing$ interfaces), which yields ~ 60 disease genes with multiple interfaces. Furthermore, only a fraction of those genes have SNPs within the interfaces. Such a small set of known disease interfaces (compared to ~ 2300 known disease genes used in the current prediction) would make any *leave-one-out* validation scarcely significant.

Nonetheless, some changes to the input dataset could make the problem viable for evaluation. For instance, a database of protein domains could be mined to determine the possible locations and number of interfaces per protein (this could eliminate the need of $\varnothing$ interfaces). Table 4.5 shows that ~ 7000 nodes have a $\varnothing$ interfaces, which could potentially derive into multiple interfaces. An increase to the number of known disease interfaces could be sufficient to make a *leave-one-out* validation significant.

The fact that drugs do not generally interact with disease genes may suggest that the prediction of drug interfaces cannot benefit from this approach. However, the prediction of a drug binding interface appears to be feasible if the protein target is known (i.e. the problem is the selection of the protein interface most likely to interact with the drug). In this case, Variant 2 could be used to diffuse the disease QWS, with no left out genes (the prediction does not intend to match the drug target to any disease gene). The final labelling will create a priority over all protein interfaces in the network. Therefore, each interface from the drug target will be prioritised towards the one more involved with the disease process. It is sound to expect that the drug would affect the interface most involved with the disease.

Considering variants from non-coding regions

Currently, the method has no way to include deleterious variants located outside the exome in the model, since the nodes and edges of the protein interface graph are defined from protein protein interactions (mined from INSIDER). However, an interface graph could be used to model a transcriptomic/regulatory network, where the interfaces are based on the binding domains of the regulatory elements. An interface graph defined exclusively from binding domains will match the considerations made in this analysis, in particular that all edges represent similar interactions. The nodes of this network will be the binding domains of transcription factors, enhancers and promoters. Notice that it is likely that no nodes will show multiple interfaces in this regulatory network, since transcription factors are not expected to contain multiple DNA-binding domains [100].

This transcriptomic network might not be sufficient to identify the extension of the disease modules, since it does not account for the genetic interactions due to protein activity. The usage of a heterogeneous interface graph, that joins the protein interface graph to the transcription interface graph, could be useful for this problem (Section 10.3 discusses possible future work in this direction). Notice that the transcription factors are proteins common to both networks, so the transcription factors will have interfaces for DNA-binding and protein interaction. Additionally, a gene promoter (a node from the transcription graph) could be considered as as an additional interface of the proteins encoded by that gene, which would integrate the relationship of protein expression into the model. The edges of this heterogeneous graph will remain connected to the same nodes as in the independent graphs.

It is unclear if the interface approach could be used to link deleterious variants to disease using other non-coding region maps, such as topologically associating domains (TADs). Notice that the usage of interfaces represent independent compartments of a well defined biological unit (e.g. a protein). Each compartment allows the biological unit an independent set of interactions to other biological units. The model is focused on the interaction between biological units, and the interfaces only group independent interactions. TADs represent areas in the DNA that interact with each other more than expected by random chance. TADs can be certainly seen as compartments, but the bigger well defined biological units (composed of TADs) are harder to define. It is also a stretch to consider the TADs as the biological units, since the interfaces would be collections physically interacting reads. It is unclear on how such a model will differ from a contact matrix (the result of a Hi-C experiment) [72].

Chapter 8

Additional Research

This Chapter presents an overview of other work and projects related to disease I have worked on during my PhD.

Sections 8.1 and 8.3 present a short overview of two collaborative projects and my contributions in those works. Sections 8.2 and 8.4 are independent works derived from those projects.

8.1 Identification of Cancer genes

This work resulted in a publication in Nature Methods called *Subclonal mutation selection in mouse lymphomagenesis identifies known cancer loci and suggests novel candidates* and is the biggest analysis of murine leukaemia virus (MuLV) induced lymphomagenesis up to date [216]. It includes 700,000 sequenced mutations from over 500 lymphoid malignancies at different stages of tumour development. The study produces a set of novel candidates for cancer. The scale of the work also allows to map low clonality inserts across multiple tumour samples to regions surrounding known oncogenes. The analysis shows evidence that rare cancer mutations may appear more frequently than expected by random chance as a subclonal mutation in late stage cancer samples. This in turn shows that subclonal mutations can provide evidence to impute rare cancer drivers.

The work is part of a long collaboration with Dr. Anthony Uren and his research groups at the MRC London Institute of Medical Sciences and the Institute of Clinical Sciences, Imperial College London.

Approach to tumour stage classification

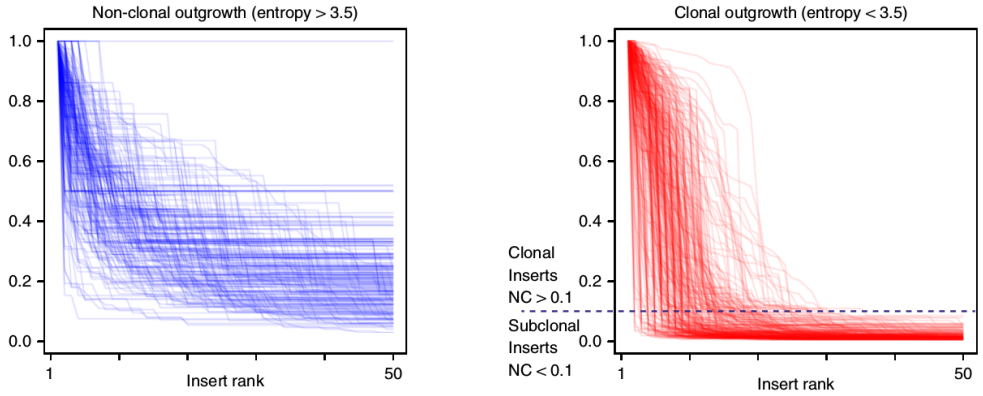
Newborn mice were infected with MuLV which cause a lifelong process of viral insertions. The accumulation of viral insertions (which can be seen as single nucleotide alterations) cause malignancies, which are studied after the mices' death. All 355 mice developed the malignancies between 42 to 300 days after infection. These lymphoid malignancies were sequenced at different growth stages to derive the profile of insertions found in the genome. Furthermore, a cohort of 166 animals were sampled and sequenced at days 9, 14, 28, 56, 84, and 128 after infection for additional reference of mutation rate.

The outcome of the sequencing process for each sample is the insertion profile at a 100kb resolution, associated with the age of the sample – i.e. the output is the sample clonality: a vector with the number of cells with insertions per common integration site (CIS). However, the stage of the tumour was not known for all samples, since the aggressiveness of the malignancy significantly varied the stage of the tumour.

My work consisted in the classification of the tumour stage from the sample clonality vectors. Manual curation of selected samples showed on one hand that the CIS clonality values for the early-stage samples are relatively low and uniform. This distribution can be expected to come mostly from random insertions while cells with the malignant mutations have not multiplied significantly. On the other hand, late-stage samples present few CIS with very high clonality values (*clonal* CIS) and most with low clonality values (*subclonal* CIS). The cells with malignant mutations are expected to multiply more, so driver mutations are expected to be included within the clonal CISs. Notice that a sample may contain more than one malignancy. The insertions associated to less aggressive malignancies are expected to appear as subclonal CISs in several samples.

The difference between the clonality vectors was quantified using the Shannon entropy [183], to allow samples to be sorted from pre-malignancy to late-stage lymphoma. In practice, only the 50 highest clonality values $c_1, c_2, \ldots, c_{50}$ are considered per sample, to limit the number random mutations analysed. The clonality vectors represent the insertion frequencies, so the normalized insertion frequencies define a probability distribution of the insertions. This probability distribution is used to calculate the Shannon entropy. Formally, c_i is transformed to p_i as:

$$p_i = \frac{c_i}{\sum_{j=1}^{50} c_j}.$$



(a) Early stage (b) Late stage

Fig. 8.1 **Clonality profiles of early and late stage tumour samples.** The profiles plot the distribution of normalised clonality values sorted from higher to lower. The few clonal CIS observed in the late stage samples suggest the presence of a cancer mutation, which replicated into many cells of the sample. On the other hand, subclonal samples may occur as passenger mutations or drivers of earlier stage malignancies. Notice that the CISs are shown in different orders on each sample, and the plot only shows a relative distribution of clonality values. *Figure excerpt taken as is from the publication [216].*

Then, the Shannon entropy E over a set of probabilities $p_1, p_2, \ldots, p_{50}$ is calculated as:

$$E = -\sum_{i=1}^{50} p_i \log p_i.$$

The entropy quantifies the spread of a distribution: it is zero when a single p_i is equal to one and all others are equal to zero, and reaches its maximum value when the probabilities are uniformly distributed ($p_i = 1/50$ for every i). The probability vector preserves the distribution of the sample clonality vector, so the entropy is expected to discriminate well the early-stage samples from the late-stage samples. Probabilities from early-stage samples are closer to a uniform distribution and therefore the samples will have high entropy values, while the probabilities from late-stage samples are closer to a spike, providing low entropy values (Figure 8.1 shows one line per sample). Further manual curation established entropy values of 3.5 or higher as late-stage samples.

Additional work

I have performed additional tasks under this collaboration which remain unpublished. Most of them are related to statistical validation of putative gene sets, in particular validation that includes PPI networks. The integration of human and mouse genes

derived on a heterogeneous interactome which includes genes from human and mouse origin presented in Section 8.2.

The network-based statistical validation consists in an over-representation analysis of known oncogenes compared to the set of putative genes and all their interactors in the PPI (*neighbourhood-1* of the putative genes). This model considers the hypergeometric distribution as the null model:

$$p = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}}$$

where N is the network size, n is the size of the neighbourhood-1, K is the number of oncogenes considered, and k is the number of oncogenes in the neighbourhood-1.

This validation was produced to identify the genes close to a high clonality CIS most likely to be an oncogene. Over 900 genes were found close to the 300 CISs with the highest clonality values across different late-stage samples. An over-representation analysis of the ~ 900 genes reported a huge enrichment for known cancer genes from the COSMIC [49] database, which provided an overall validation for the selection method.

The set of candidates was narrowed by selecting mutated genes which appeared significantly biased in one of 4 categories[1]: found more often in the top 50 CIS of late stage samples (compared to early stage), preferential insertions in the forward or the reverse strand (strand specificity), developed preferentially into T cell or B cell lymphomas (lymphoma subtype specificity), or if the mutated gene developed lymphoma significantly more rapidly in the wild type or BCL2 transgenic (genotype specificity). A gene that fits any one of this criteria is considered as a valid candidate. Therefore, the 420 genes that fit at least one criteria are set as candidates.

Notably, 47 of those candidate genes are part of a high quality consensus list of 291 known cancer genes [52]. Out of those, 21 were verified as subclonal mutations with a late stage bias. This shows that subclonal mutations can provide statistical evidence to identify cancer drivers with insufficient clonal mutations [216].

8.2 Human Mouse Interactome

This project describes a general procedure to combine protein-protein interaction networks from different organisms into a single heterogeneous interactome [17]. The

[1] The bias of a gene towards a category is evaluated with an over-representation analysis of the gene in a particular category. No network components are involved in this evaluation.

approach is suggested to be useful for the analysis of genetic experiments performed in organisms other than the intended target.

This pipeline was used to construct a human-mouse network which was used during the research stages of the lymphomagenesis project. This network served to contextualize the candidate mouse proteins in a human interactome. The publication intends to share a working pipeline, and to discuss some common issues found when mapping proteins from different databases.

The publication includes a detailed mathematical notation and formal criteria to map gene synonyms in databases with possibly conflicting identifiers. These details are skipped here for the sake of simplicity, and the mapping between different databases is considered to be seamless. Nonetheless, the results strongly suggest that the usage of gene symbol synonyms produces erroneous protein matching.

Producing a combined interactome

The first step to combine two organism graphs $G_a = \langle V_a, E_a \rangle$ and $G_b = \langle V_b, E_b \rangle$ is to obtain the set of equivalent vertices. The equivalent vertices merge all the homologous proteins found in the pair of organisms – i.e. vertices $v_a \in V_a$ and $v_b \in V_b$ are merged v_{ab} if v_a is homologous to v_b. Notice that two vertices of the same organism a can be merged into a single combined vertex if they are homologous to the same vertex of the other organism b. The combined vertices V_{ab} can be considered to belong to different homology classes.

The edges of the combined interactome E_{ab} are aggregations of the edges of the independent graphs E_a and E_b. If the vertices of an organism have an edge, the combined vertices will also have an edge $((v_a, u_a) \in E_a \vee (v_b, u_b) \in E_b \rightarrow (v_{ab}, u_{ab}) \in E_{ab})$. This yields edges with three possible evidence types: evidence from a, evidence from b or evidence from both. This concludes generation of the the combined graph $G_{ab} = \langle V_{ab}, E_{ab} \rangle$.

In particular, the human-mouse interactome is produced from the BioGRID v3.2 database [24], which includes protein-protein interaction networks for human and mouse. Furthermore, BioGRID includes a some proteins human proteins in the mouse interactome, and conversely mouse proteins in the human interactome. This mapping is used as the homology mapping when possible. Additional homology pairs are mined from the Human-Mouse homology database [45] provided by the Mouse Genome Informatics (MGI) project.

The final human-mouse interactome consists of 17,644 vertices and 169,458 edges. Notably 6,139 out of 6,824 of the mouse genes mined from the 2014 BioGRID database were homologous to a human gene. Considering the increase in size from

the mouse interactome, the final combination represents an increase of over 150% in the number of vertices, and an increase of over 430% in the number of edges. This number comes much closer to the expected 25,000 genes in a mouse interactome [61]. Notice that recent releases of BioGRID have considerably increased the size of its mouse interactome, so this procedure is no longer needed to complete that network. However, the approach is still useful for other organisms with smaller known interactomes.

A short lesson on synonym mapping

Genetic databases frequently identify genes or proteins by "standardised" gene symbols and proprietary identifiers [24, 31, 117, 226]. To match heterogeneous updates in nomenclature, the PPI databases frequently contain synonyms to a main symbol. However, these synonyms are not *equivalent* names for the protein in the mathematical sense.

In maths, an equivalence relation $\equiv$ is a binary relation which is reflexive ($a \equiv a$), symmetric ($a \equiv b \rightarrow b \equiv a$) and transitive ($a \equiv b \wedge b \equiv c \rightarrow a \equiv c$). This relation can be used to build the equivalence class of an element $[a]_{\equiv}$, which is defined by elements related to a by $\equiv$.

Considering gene symbol synonyms as equivalences, finding the set of all equivalence classes in BioGRID yields a class containing more than 2,500 genes. These genes are definitely not equivalent (or homologous) in a biological sense.

Gene synonyms are not transitive.

8.3 Prediction of drug cocktails for Chagas Disease

This project consists in the prediction of drug cocktails using FDA approved drugs, with putative effect against the *Trypanosoma cruzi* (*T. cruzi*) during the chronic phase. The *T. cruzi* is vector based parasite, that produces Chagas disease and is endemic to Latin American countries. The disease has two phases: acute and chronic. There are two drugs effective against the disease during the acute phase (benznidazole and nifurtimox), however this phase is often asymptomatic and the disease remains undetected until the chronic phase. While some individuals never develop symptoms, the chronic phase affects the digestive system, nervous system, or heart and may result in sudden death.

The computational analysis is divided into two approaches to produce sets of putative drugs, a multi-objective approach to select the drug cocktails and *in vitro* experimental validation of the cocktails. This work derived in an IEEE Xplore

publication called *Drug cocktail selection for the treatment of chagas disease: A multi-objective approach* [198].

The project is collaboration with research groups from the Universidad Católica "Nuestra Señora de la Asunción" (UCA) and Centro para el Desarrollo de la Investigación Científica (CEDIC) from Paraguay.

Overview

The first approach selects drugs based on homology and includes different layers. The organism layer selects drugs designed for organisms close to the *T. cruzi*. The organisms are considered close based on an evolutionary distance in the tree of life and similarity of the 18S rRNA. The metabolic pathway layer selects drugs which target enzymes homologous to those found in *T. cruzi* pathways. The pathways were inferred using the Pathway Tools suite [84] and mined from KEGG [82].

The second parallel approach produces machine learning models to predict an antitrypanosomal drug effect. The models are trained on results of small molecule screening studies and include chemical features and feature-connectivity fingerprints [169] to build a Quantitative Structure Property Relationship (QSPR) model [201].

The selection of drug cocktails is considered a multi-objective approach, which integrates the biological concepts from the homology approach to the output of the machine learning model. The cocktail is expected to maximise the number of enzymes targeted, the number of pathways targeted, and the percentage of enzymes covered per pathway. All these measures are intended to increase the likelihood of disrupting essential genes or processes in the *T. cruzi*. Notice that the multi-objective approach naturally deals with the possibly conflictive objectives. For instance, given a reasonable cocktail size (e.g. less than 6 drugs), an increase in number of pathways targeted is likely to reduce the percentage of enzymes covered per pathway.

Due to constraints for *in vitro* verifications, about a dozen cocktails were tested so far prioritising new candidate drugs. While these showed modest effectiveness against the disease, several drugs generated from the machine learning approach were included in previous studies against Chagas [94]. This suggests that the methodology is sound and that it may produce results of biological significance.

My work in this collaboration consisted mainly in guiding graduate students from the UCA, in particular for the homology approach, and producing the drug combinations.

8.4 Characterisation of Drug Modules

This is a personal work [18] which considers a complementary characterisation of disease by analysing drugs used for their treatment. The working hypothesis is that drug targets should interact with the disease module in order to produce an effect to treat the condition. While ideally drug targets would be part of the disease module, drug mechanisms of action are incredibly heterogeneous.

Some drugs can be reasonably expected to interact with the disease module, such as supplements, which are designed to replace missing nutrients or compounds. Therefore, these drugs would be part of a metabolic pathway related to the condition. However, drugs are far more frequently designed to treat the symptoms of a disease, and may incur in complex interactions to produce the desired effect [73]. Common examples are painkillers (i.e. anti inflammatories and opioids), anti-acids (i.e. histamine antagonists, proton pump inhibitors), and blood thinners. Among these drugs, opioids are particularly not expected to belong to disease modules, since they produce an effect by blocking signalling pathways [73].

These complex interactions prompted the usage of network analysis to understand the role of drug targets and disease. Drug targets are rarely protein products of essential genes. However, they still target proteins inside topological modules. The average drug target degree is above the network average and below the essential gene average[2] [227]. These properties proved helpful for the identification of functional modules affected by the drugs other than the intended effect, which in turn yielded applications for drug repurposing [70].

This work develops a general basis to analyse the phenotypic and genotypic relation between drugs and heritable diseases by proposing *drug modules*. Since drugs and diseases are *a priori* incomparable, a disease-drug category mapping is initially produced. The disease categories are used roughly as disease families, and define disease modules [90, 139, 148], while the drug categories yield a novel lax approximation to the drug target counterpart of disease modules – i.e. drug target modules.

The analysis has two components, a characterisation of drug target modularity by the assessment of modular features, and later a comparison between drug and disease category modules (see Section 4.2.2 for more details). Target locality and functional similarity of drug targets to quantify their specificity and possible side effects, which can be exploited for drug repositioning.

[2]The essentiality of a gene is known to correlate to the degree of the protein it encodes. In average, essential proteins have significantly higher degree than the network average [227].

Results

Category mapping

This is a hand curated association between the Goh *et al.* disease categories [57] and ATC categories [146]. While many categories can be found in both Goh and ATC, some are not included in comparable groups, thus DrugBank's pharmacological action categories [219] are used as fall-back whenever the Goh category did not match an ATC category. The entire mapping and a summary of the number of elements per category can be found in the Appendix B.2.

PPI analysis

The average distance between all disease genes is 3.45, while the average interactome distance is 3.58. Drug targets appear to be somewhat closer than random genes, with an average distance of 3.25. Welch's t-test confirms that the distance distribution from drug targets is significantly different from a random choice in the interactome (p-value $< 1.0 \times 10^{-300}$). The slight modularity suggests that there are some functions which are particularly good as gene targets. Figure 8.2 shows that intra-category proteins are not in general closer than inter-category proteins, however Cancer and Endocrine drug targets are notably close. Looking at individual rows, Bone, Developmental, Gastrointestinal and Immunological modules appear to be located in particular areas of the network, as their intra-modular distance is smaller than their inter-modular distance. These results seem to indicate that most drugs do not affect specific modules in the network, but rather target proteins which have an indirect effect on the disease. This is supported by the fact that many drugs treat the symptoms of the disease rather than the causes (e.g. painkillers and anti-inflammatory drugs).

Taking the known genes from the drug categories as the gold standard, gives an average Jaccard coefficient of 0.027 for all drug and disease categories (this is comparable to Ordinal predictions from Section 4.2.2). Figure 8.3 illustrates the pairwise overlap between drug and disease categories (notice that only the diagonal is used to calculate the average Jaccard).

Functional analysis

The GO over-representation analysis yields a high amount of signalling, transport and binding functions for most categories, suggesting that drug compounds target peripheral mechanisms to produce their effect. Signalling pathways are known to coordinate cell responses, while binding and transport functions mediate the

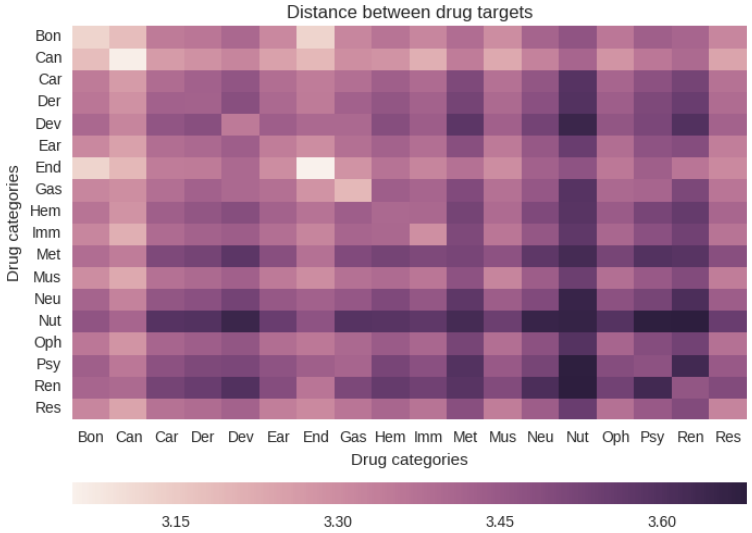


Fig. 8.2 **Heatmap of drug target distance for category pairs.** Colors indicate the average distance between all drug targets from category *A* to targets from category *B*.

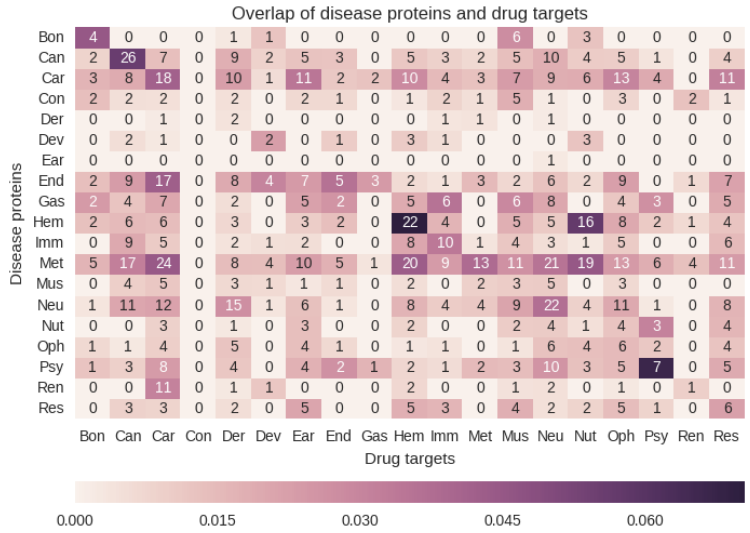


Fig. 8.3 **Heatmap of the Jaccard coefficient between disease proteins and drug targets by category.** Colours are based on the Jaccard coefficient, while labels indicate the actual number of common elements in the categories.

availability of substances in the organism. These mechanism of action seem to control the downstream effects of disease genes, which supports the idea that drugs frequently target symptoms rather than causes of the disease. The significance analysis clearly discriminates between few drug categories which target specific functions (in particular Bone, Developmental, Gastrointestinal and Renal), and most drug categories which target a broad spectrum of functions (in particular Cancer and Cardiovascular).

These results can expected from the PPI analysis and known trends of drug development (i.e. Cancer drugs are known to have large amounts of side effects and are used as an extreme measure [46]). The notable exception is the specificity of Renal drug targets, however the small size of the module could explain a bias in the PPI experiment (only 10 drug targets were found in HPRD).

8.5 Network Visualisation

The massive amount of data involving biological networks, makes the presentation of the information in a useful way (and if possible, grant a user friendly interaction) one of the major concerns of Systems Biology. Common examples of such networks are PPI networks with 9 thousand to 20 thousand proteins, and 40 thousand to 4 million interactions [160, 144], 4 thousand disease genes [125], 4 thousand drug targets [219], and other networks of comparable sizes [9].

Handling the networks as a big data collection is most useful for automated analysis of biological systems, since adding evidential support offers machine learning techniques bigger test sets to make better predictions. On the other hand, expert analysis is rarely efficient on such hefty data collections. Normal approaches of handling big data collections would use both types of analysis, automated analysis prioritise data for a detailed expert analysis, so the likelihood of encountering significant results in the study would increase.

During the past years I have worked with lab members to build a couple of network visualisation tools, mainly focused to aid the research of disease. The first tool called LanDis, which is a freely available web-based interactive application www.paccanarolab.org/landis that allows domain experts, medical doctors and the larger scientific community to graphically navigate the landscape of human disease similarities. This particular project is ready and a working manuscript going through an extensive publication process.

The second tool is developed for an ongoing research project, focused on the effects of a particular drug for the treatment of Multiple Sclerosis (MS). This tool

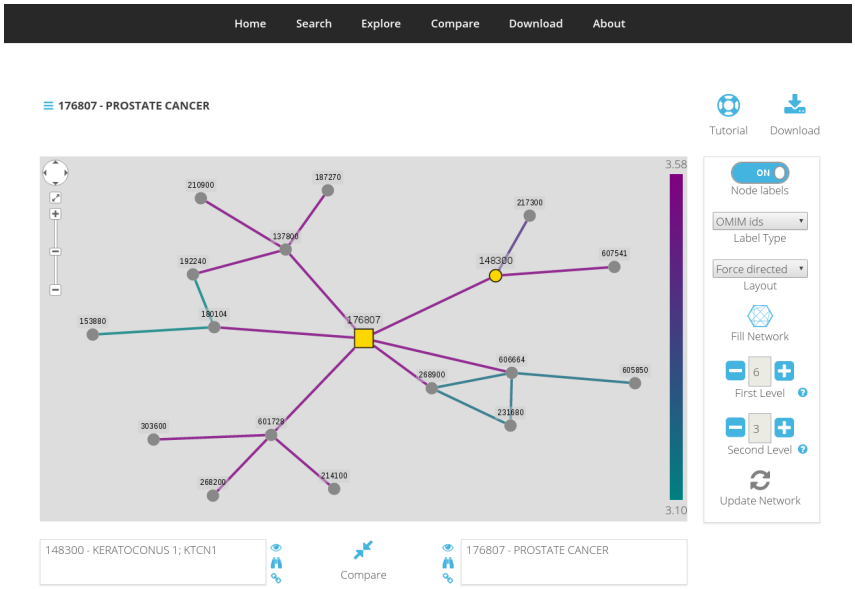


Fig. 8.4 **LanDis: The Disease Similarity Landscape Explorer.** LanDis allows the exploration of the disease similarity landscape based on disease phenotype, and offers the possibility of detailed pairwise disease comparisons to analyse the factors of the similarity.

allows the visualisation of MS proteins, their neighbourhoods, and their relationship with proteins of significantly similar diseases or proteins of relevant biological pathways involved in the disease. As this is an ongoing project, we cannot make this tool public and it's not yet clear if this tool will be available for public use in the future.

Although tools development is not a priority of the research, these tools are still valuable contributions for the Biological Systems community.

Chapter 9

General discussion and conclusions

This chapter focuses on the contextualisation of the different outcomes of this work within the field of network medicine. Since this work includes a compilation of multiple correlated methods and projects, the particular contributions are presented in their corresponding chapters. The ideas presented here are related to general concerns of the discipline which are worth revisiting, and the nuances offered by this work.

Section 9.1 presents an overview of the challenges in the prediction of genes for molecularly uncharacterised diseases. It analyses possible reasons for the reduced performance on prediction, and how these reasons match the different testing configurations.

Section 9.2 analyses several factors that contribute to the performance of Cardigan and the variant on protein interface networks. The analysis includes a comparison between the performance in the predictions of entirely novel genes and genes associated to multiple diseases, the trade-off given by the usage of different networks, vertex coverage, weighted edges, edge density, and different diffusion methods on the ECLAIR network.

Section 9.3 serves as a summary of the contributions of about network module principles presented in this work after a short overview of the field.

Section 10.2 presents a novel empirical framework to assess the expected performance from Cardigan, and shows a procedure to select diseases where a high performance can be expected.

9.1 Predicting on uncharted diseases

The existence of millions of SNPs and other mutations give a nearly unending amount of variance in the human genome. Due to the computational and financial

costs associated to whole genome studies, exome studies are frequently considered to study human disease. The human exome contains between 12 thousand to 35 thousand SNPs, out of a total estimate of 500 thousand SNPs. In average, 40% of the mutations are synonymous, 59% are non-synonymous and 1% are nonsense [29, 224]. This abundance of alleles renders the identification of causal genes for rare diseases a dismal task.

Network based methods facilitate the task by narrowing the number of candidate genes, which allows for a greater statistical significance of the mutations. However, they work as supervised or semi-supervised learning approaches, which require initial data to be fed to the network [235]. So far, the only network methods able to predict genes for uncharted disease without sequence data use disease phenotype to create initial labels [135, 206] (see Chapter 3). However, they operate with matrices calculated with outdated phenotype data [205], and are proven unable to predict any genes which became associated to a disease in OMIM for the past 4 years (see Section 6.2.1).

Cardigan allows the prediction for uncharted diseases by providing a seamless integration with OMIM databases to obtain disease gene associations. The disease phenotype is also obtained through OMIM databases, and is calculated as the semantic similarity between the MeSH terms describing the diseases found in scientific publications (see Section 4.3.3). Initial data for the prediction consists in weighted labels from genes belonging to diseases similar to the query, and the weights are given by the similarity of the disease they belong to and the query disease. This initial data already takes the shape of a putative disease module for the query disease (see Chapter 5). The labels are finally diffused through an interactome to produce a gene prioritisation as a ranking for all genes in the network. Cardigan uniquely is able to predict recent OMIM associations and vastly outperforms other state-of-the-art methods on several testing environments (see Section 6.2).

The presented disease gene prediction evaluation consists of two different variables: knowledge of molecular basis and cause of the missing target. Charted diseases have some known molecular basis while uncharted disease have no molecular basis, in both cases a single missing gene is predicted. *Time-lapse* experiments consist in using 2013 OMIM data is taken to predict genes which were associated to OMIM in the 2013-2017 period. *Leave-one-out* experiments synthetically remove one known gene at a time, and predict it back using 2017 OMIM data. Figure 9.1 compares the performance of Cardigan under the different testing conditions.

It is reasonable to think that the fluctuation in performance for the different environments depends exclusively on the amount of data available. The significant decrease in performance between the *time-lapse* charted and uncharted tests is

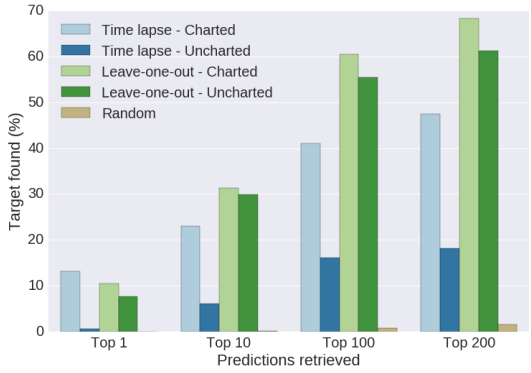


Fig. 9.1 **Cardigan's performance for different testing environments.** Percentage of disease genes found in the predictions vs. the number of predictions retrieved. All predictions are made using HPRD.

expected as uncharted diseases are likely to have fewest scientific publications and are the most difficult to predict. However, *leave-one-out* uncharted experiments suggests a dramatically different scenario as it appears to produce 80%-90% of the accuracy of a *leave-one-out* charted prediction.

The difference in accuracy between *time-lapse* and *leave-one-out* experiments can be explained as the removal of the gene association does not discard other information included in the method prediction. For instance, the discovery of the gene could lead to a better phenotype characterisation, enhancing the phenotype similarity matrices. Another difference is that studies for complex disease frequently find entire new sets of genes associated to a trait [63], so finding one gene missing out of a coherent module can be expected to be easier than finding the entire module. In this way, *time-lapse* experiments reflect real scenarios where missing genes are predicted, while *leave-one-out* experiments reflect a scenario where diseases have many scientific publications where a single causal gene is yet to be found. Thus, *time-lapse* testing approximates the average performance of the method on real studies, while *leave-one-out* testing approximates the upper bound in performance expected of the method.

Unfortunately these considerations are frequently disregarded in many publications of the field, which only include *leave-one-out* testing environments [3, 50, 55, 90, 107, 111, 139, 148, 165, 184, 186, 206, 222, 223, 231].

Analysis of particular time-lapse examples

While *time-lapse* experiments are better characterisations of the disease gene discovery process, some unknown associations are somewhat artificial due to the way in which OMIM annotates diseases. All the following examples presented no genes

associated for the disorder in the 2013 OMIM database, and had an association made available sometime before the 2017 OMIM database.

- Glass Syndrome (MIM:612313) is characterised by mental retardation, facial abnormalities, microcephaly, scalloped skin [56]. Multiple case studies revealed a linkage with the SATB2 gene, at least for some of the presented clinical features [170, 203], however only by December 2013 a study was published associating the syndrome through and underlying genetic interaction between SATB2 and the UPF3B gene [104]. This gene was predicted as the second prediction by Cardigan using the HPRD interactome.

- Bruck Syndrome (MIM:259450) is characterised by contractions, growth deformities such as pterygia, webbing and clubfeet, and multiple fractures from childhood. Early documentation of the rare syndrome dates from 1969 [155]; while multiple papers analysed particular patients [15, 126]. Later, a chromosomal region appeared to be identified by 2010 [181]. A posterior analysis, identified the same mutation (nucleotide deletion) in both alleles of the FKBP10 gene on September 2013 [11], which is posterior to the publication of the old database used. Cardigan was able to predict FKBP10 as the fifth prediction with the FUNCOUP interactome.

- Abruzzo-Erikson Syndrome – ABERS (MIM:302905) was described as a new syndrome of cleft palate, coloboma, hypospadias, deafness, short stature, and radial synostosis in 1977 [1]. The publication which associated the TBX22 gene to the syndrome was published on April 2013, but was not included on the 2013 OMIM database used for this prediction. No documentation curated by OMIM linked the TBX22 gene to ABERS, however Cardigan was able to predict TBX22 as the tenth prediction using HPRD.

These examples come from an experiment not tractable for the other methods analysed, however some information about their molecular basis is known. The Glass Syndrome case included several papers which suggested the gene to be associated, but the gene remained to be annotated in OMIM. The Bruck Syndrome had cases analysed through a long time and suggestions of chromosomal areas where the gene was later found. And finally, ABERS had no papers linked to genetic positions. However, the gene was associated to a dummy disease related to cleft palate in the 2013 OMIM. While all of these cases are computationally hard, experts in the field are expected to know about these putative genes limiting the novelty of the predicted association.

Notice that after a literature review all the disease examples presented for Cardigan in Table 6.2, contain genes not found in publications related to the examples older than the 2013 OMIM database release. That suggests that Cardigan indeed contributes to find genes for real molecularly uncharacterised diseases.

9.2 Factors that influence performance

9.2.1 Novel disease genes

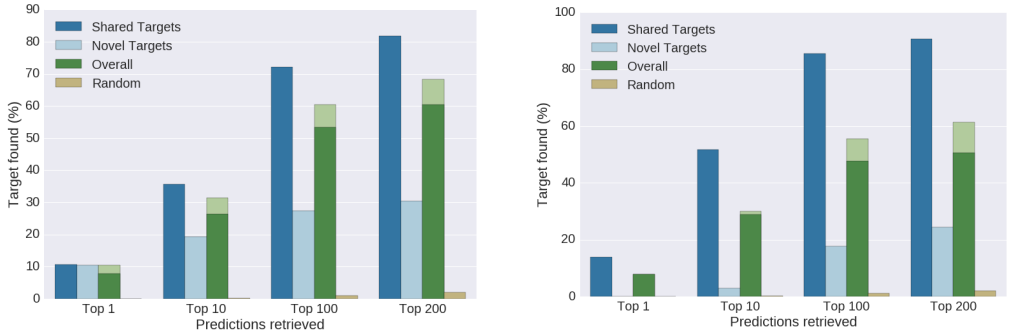
Given the fact that Cardigan has a strong baseline prediction given by the shared genes of other diseases (i.e. the Ordinal prioritisation), it is interesting to decompose the analysis into the prediction of genes found in OMIM or outside the database.

The first question is how many genes in Cardigan's output are known for other diseases, but are not known to be associated with the query. In average, 28% of the top 200 predictions are genes not known to be associated with any disease on *time-lapse* experiments, and 26% for *leave-one-out* experiments. The fact that more known targets are predicted is expected since the training set contained a similar distribution of genes with known and unknown associations as targets.

The second question is the difference in performance while predicting genes known to be associated with multiple OMIM diseases (i.e. *shared targets*) or genes not known to be associated with other OMIM diseases (i.e. *novel targets*). From the 769 OMIM disease gene associations for diseases with two or more genes predictable in HPRD (*leave-one-out* charted), 201 (26.1%) are genes associated to a single disease in OMIM. Analogously, 1,442 (44.3%) of the 3,252 diseases with one gene association (*leave-one-out* uncharted) are synthetic novel targets. Figure 9.2 shows that predictions are much harder for novel targets, however that performance is still significantly higher than what is expected by random chance. Notice that some scenarios are more favourable for the prediction of novel targets, for instance the performance for the first prediction of charted diseases is equal for novel and shared targets.

These experiments show that Cardigan is not only able to predict genes for uncharted diseases, but also provides novel gene candidates.

The results on *time-lapse* experiments are also cited for the sake of completeness. Charted *time-lapse* experiments contain 8 (13.1%) genes not associated to any disease in the 2013 OMIM release and none were predicted among the top 200 predictions. Uncharted *time-lapse* experiments contain 80 (53.7%) genes not associated to any disease in the 2013 OMIM, and 2.5% of them were found among the top 100 and 200 predictions. These results continue to show the difference in performance between



(a) *Leave-one-out* charted (b) *Leave-one-out* uncharted

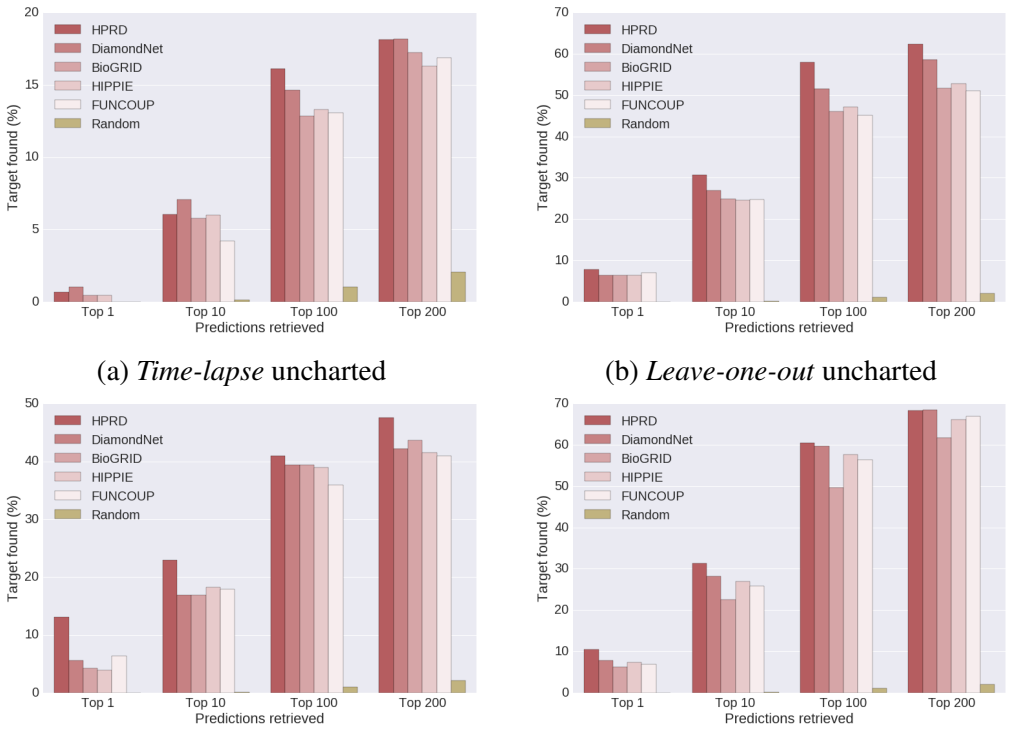
Fig. 9.2 **Cardigan's performance at predicting novel and shared disease genes.** Percentage of disease genes found in the predictions possible per network vs. the number of predictions retrieved on HPRD. The bars show the performance normalised for the amount of predictions available for each dataset. *Shared targets* are genes known for other diseases and start with a seed for the diffusion. *Novel targets* are not associated to any disease (synthetically removed from the query) and do not have a seed. The stacked *Overall* bar is the performance on the entire dataset, the pieces in the stack are highlighted to reference the contribution of shared (dark) and novel (light) targets. (a) Performances for a leave-one-out testing for diseases with two or more known genes in 2017. (b) Performances for a leave-one-out testing for diseases with a single known gene in 2017.

time-lapse and *leave-one-out* experiments. Furthermore, it suggests that a good option of follow up development is enhancing the prediction of entirely novel disease genes.

9.2.2 Trade-off in real networks

The network medicine paradigm relies on the fact that these biological networks encode the complex interplay between cellular components, into a simple yet powerful data representation. While in depth genetic research (i.e. protein interactions, drug affinity or function) is vital to characterise a disease, it frequently derives to knowledge specific to the phenotype under observation or expensive research pipelines [141, 173]. These studies, however, provide data which allows the construction of biological networks sprinkled with high quality information. Modern computational methods gain insight from the initial information by exploiting local and global patterns, and produce inferences with reasonable support. Naturally, the inference quality hinges on the amount of known data, network characteristics and critically the inference method.

It is essential to recognise that binary representations may not be ideal to depict many biological phenomenons. It may seem intuitive to pursue a binary model for



(a) *Time-lapse* uncharted

(b) *Leave-one-out* uncharted

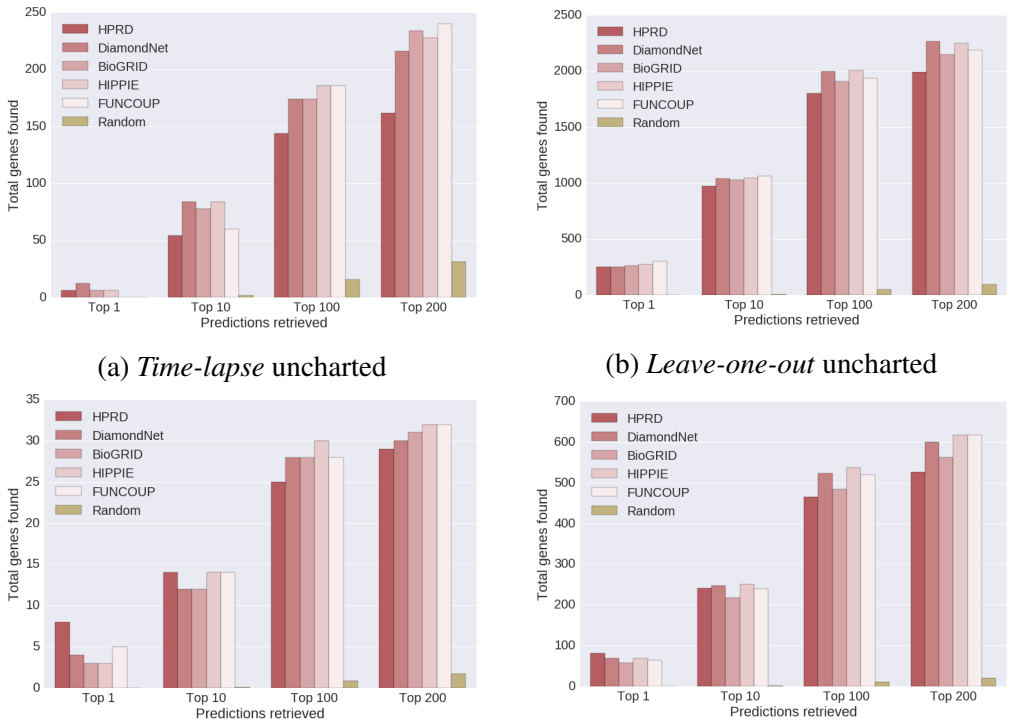
(c) *Time-lapse* charted

(d) *Leave-one-out* charted

Fig. 9.3 **Cardigan's performance on different networks.** Percentage of disease genes found in the predictions possible per network vs. the number of predictions retrieved. (a) Performance for diseases which were uncharted in 2013, but were charted in 2017, measured on different PPI networks. (b) Performances for a *leave-one-out* testing for diseases with a single known gene in 2017 on HPRD. (c) Performance for predicting the genes that charted diseases have acquired between 2013 and 2017. (d) Performances for a *leave-one-out* testing using 2017 data.

protein-protein interactions, where a pair of proteins either interacts or not. However, proteins have complex interaction preferences with other proteins, which are partially explained by the interaction affinity. Further complexity can be seen as the affinity between a pair of proteins is not static, and may change due to interactions with other proteins or compounds [85]. While both binary and weighted PPI networks lead to biases in the recognition of network modules if they present incomplete or inaccurate data [238], binary networks are intrinsically bound to contain both.

The compilation of high quality data and predictions comes at the cost of completeness. This can be seen in PPI databases such as the sparser hand curated HPRD [160] in contrast with the dense and experimental FUNCOUP [144, 178], in disease gene prediction methods the highly specific Navlakha 1 and Navlakha 2 variants [139] over the generalist Oti *et al.* method [148], or in protein function predictions as a trend among participating methods in the CAFA competition [79, 164].



(a) *Time-lapse* uncharted

(b) *Leave-one-out* uncharted

(c) *Time-lapse* charted

(d) *Leave-one-out* charted

Fig. 9.4 **Cardigan's total recall on different networks.** Total disease genes found among the top predictions vs. the number of predictions retrieved. (a) Performance for diseases which were uncharted in 2013, but were charted in 2017, measured on different PPI networks. (b) Performances for a *leave-one-out* testing for diseases with a single known gene in 2017 on HPRD. (c) Performance for predicting the genes that charted diseases have acquired between 2013 and 2017. (d) Performances for a *leave-one-out* testing using 2017 data.

Cardigan is designed to handle binary, weighted, experimental and inferred interactomes, and allows the user to run the predictions with their network of preference. Figure 9.3 shows how Cardigan handles different networks on multiple arrangements for disease gene prediction.

While the performance decreases on the larger networks (see the lighter bars on Figure 9.3), they are expected to be harder predictions. As more nodes are added to the network the number of true positives stays the same (i.e. the single target being predicted), while the number of true negatives grows (all other nodes).

However, Figure 9.4 shows that the total number of genes found among the intervals improves, as more disease genes are contained within the interactomes.

Figure 9.3 shows that the average performance of the method decreases for the bigger networks, however Figure 9.4 shows that the bigger networks contain indeed more information, and serve to identify more disease genes than the smaller networks.

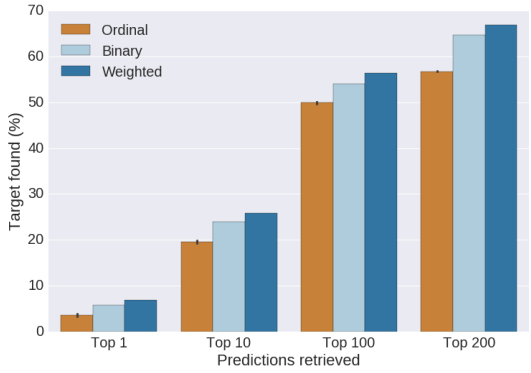


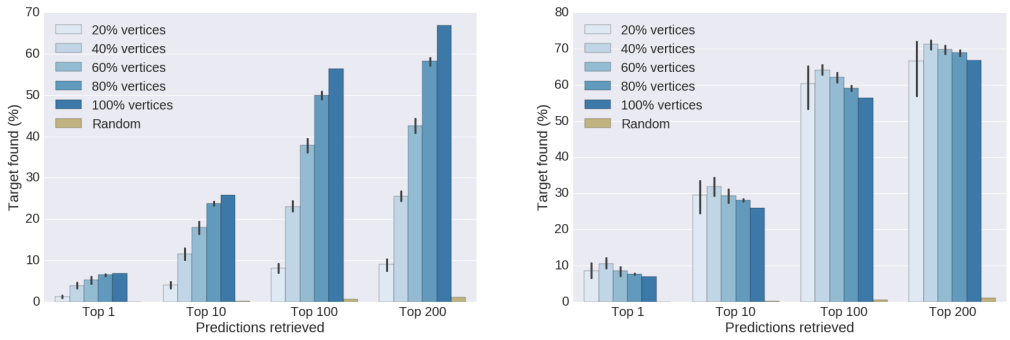
Fig. 9.5 **Comparison of performance between binary and weighted networks** Performance of Cardigan for a leave-one-out charted testing on FUNCUP using the edge weights or binary edges.

The difference between the figures is particularly large for *time-lapse* uncharted targets. Notice that BioGRID and FUNCOUP have roughly double the amount of genes of HPRD (see Table 4.3), so they include more disease genes annotated in OMIM (BioGRID and FUNCOUP cover $\sim 70\%$ of the OMIM genes and HPRD $\sim 50\%$). The drop in average performance seems reasonable, considering that there is a much larger number of negative samples in each prediction. The criteria to select one network over the other is likely to be a complex issue and cannot be answered in a general way; the final user of the method must deal with the inescapable trade-off between precision and recall.

9.2.3 Contribution of weights, coverage and density

The following experiments are designed to weigh the contribution of the factors that contribute to the network performance: edge type (binary or weighted), protein coverage (amount of vertices in the network) and edge density (amount of edges in the network). This analysis fixes the network to FUNCOUP [178], and modifies vertices or edges to test the different features. Furthermore, the experiments evaluate the performance in *leave-one-out* tests for diseases with two or more genes associated in OMIM.

The first experiment compares weighted and binary edge types (Figure 9.5). The binary edges were produced by setting all connected edges in FUNCOUP with a weight of 1. While different thresholds could be tried, all the edges available in FUNCOUP are already considered above a baseline confidence threshold [178]. The baseline performance for this comparison is set by Ordinal, which can be considered as the network with no diffusion. Figure 9.5 shows that the difference in performance is particularly sensitive for the top results, where they drop nearly by 50% in the first



(a) Vertex sensitivity

(b) Normalised vertex sensitivity

Fig. 9.6 **Comparison of performance by variation of node coverage.** The experiments consist in Cardigan predictions on synthetic *leave-one-out* tests for OMIM charted diseases, using the FUNCOUP network and percentages of vertices are kept in the network. The error bars show the standard deviation of 10 random samples. (a) Shows the percentage of possible targets from the complete network being retrieved. (b) Normalises the measure to show the percentage of targets still available in the network to be found.

prediction, and by 20% in the top 10 predictions. However, in average the prediction is halfway between the baseline and the usage of weights.

The second experiment compares the sensitivity to the network coverage (Figure 9.6). The baseline given here is a random selection of genes, as there are no changes intrinsic to the diffusion process. Figure 9.6a shows that the recall is extremely sensitive to the amount of nodes available in the network, as expected. However, Figure 9.6b shows that taking into account the available results, the method maintains a stable performance.

The third experiment compares the sensitivity to the network density (Figure 9.7). The baseline is Ordinal, to show a case with no diffusion. Figure 9.7 shows a non-linear relation between the number of edges dropped and the performance, and a big variance in performance on high edge removal cases. The removal of the first 20% of edges results in a small loss of performance, which could be explained by the density of the network and the inferred nature of the edges. The perturbation is massive when 80% of the edges are removed, causing frequently a lower performance after diffusion. The fluctuation in performance is likely to respond to changes in the connectivity in the network, and suggests the existence of essential connected components network.

While most experiments confirm the notion that more information is useful, the edge density suggests that some very incomplete datasets may skew the predictions and hurt the performance. In these experiments, the effects of using of binary edges equates to a drop of about 50% edges throughout the top 200 predictions. The

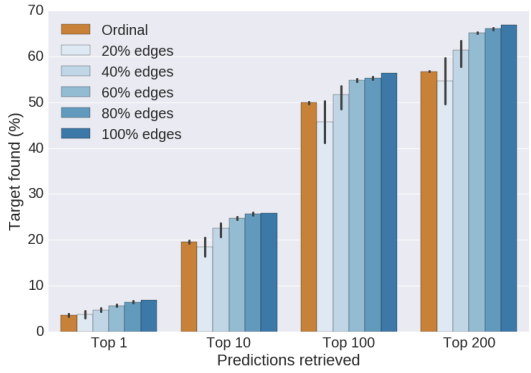


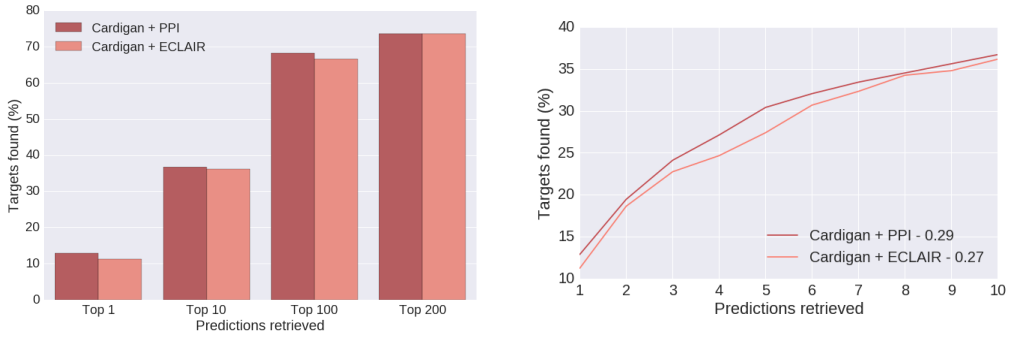
Fig. 9.7 **Comparison of performance by variation of edge density.** The experiments consist in Cardigan predictions on *leave-one-out* tests for OMIM charted diseases, using the FUNCOUP network. The error bars show the standard deviation of 10 random samples. (a) Different percentages of vertices kept in the network. (b) Different percentages of edges kept in the network.

vertex coverage shows a different effect pattern, which appears roughly as a linear fluctuation in performance.

9.2.4 Contribution of network interfaces

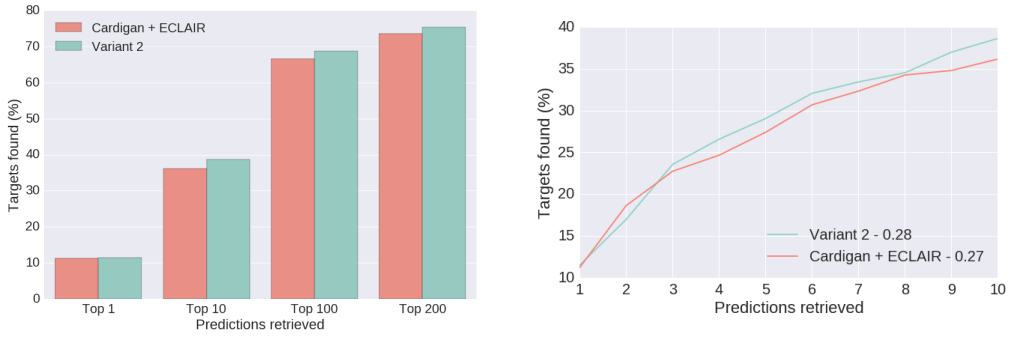
It appears that under a consistent testing scheme, the addition of data always produces better results. However, the improvements can only be seen when the system is prepared to handle the data. Chapter 7 presented the benefits of adding data (in the form of interfaces) to a PPI. It might seem intuitive that Cardigan would benefit alone from its usage. However, Figure 9.8 shows how the usage of the network with interfaces decreases the performance of Cardigan if the diffusion method stays the same.

The results of the evaluation can be better understood by taking into account the difference between the INSIDER PPI and the INSIDER ECLAIR graphs (construction shown in Section 7.1.1, and summaries shown in Tables 4.4 and 4.5). It is interesting that the INSIDER ECLAIR graph has only 14% more nodes than the INSIDER PPI graph. In fact, 36% of the proteins have no high quality interfaces (which appear as ∅ interfaces in the graph), and 51% of the proteins have a single high quality interface. Therefore, the entire difference between the graphs is accounted by 13% of the INSIDER proteins with 2 or more interfaces. While Section 7.1, addresses that a possible drawback of using an interface graph is the reduction in connectivity, the small amount of multi-interface proteins currently predicted greatly diminishes this effect. In fact, only 0.1% of the INSIDER proteins have no single interface in the largest connected component of the ECLAIR graph.



(a) (b)

Fig. 9.8 **The consistency method on a PPI with interfaces.** Comparison in performance by using the INSIDER-PPI and INSIDER-ECLAIR networks (see Section 7.1.1) for the standard Cardigan diffusion method (see Section 6.1). The experiments are *leave-one-out* tests for ClinVar diseases with more than one associated protein – i.e. Charted tests. (a) Percentage of disease genes found in the predictions vs. the number of predictions retrieved. (b) Normalised ROC for the first 10 false negatives.



(a) (b)

Fig. 9.9 **Comparison diffusion methods on a PPI with interfaces.** Cardigan uses the consistency method (see Section 6.1) for diffusion and Variant 2 uses an interface directed proposal (see Section 7.1.4). The experiments are *leave-one-out* tests for ClinVar diseases with more than one associated protein – i.e. Charted tests. Both algorithms use the INSIDER ECLAIR network (see Section 7.1.1). (a) Percentage of disease genes found in the predictions vs. the number of predictions retrieved. (b) Normalised ROC for the first 10 false negatives.

Recall that Figure 9.8 shows that Cardigan loses performance when expanding the protein graph into the protein interface graph; this suggests that the Zhou diffusion cannot account for the composite nature of the protein interface nodes. Figure 9.9 shows that a diffusion method designed for protein interface graphs (Variant 2) is in fact better at using the INSIDER ECLAIR graph. The difference in Figure 9.9 is comparable to using 60% or 100% of the network edges from Figure 9.7, which speaks for the benefits of an appropriate diffusion method. Figure 7.4b (from Section 7.2) showed that using Variant 2 on the protein interface graph produced better results than the Zhou diffusion on the protein graph for the top 10 predictions and onwards. The difference in Figure 7.4b is comparable to using 80% or 100% of the network edges from Figure 9.7, which speaks for the information gained by considering the protein interfaces.

Overall, this analysis shows that the availability of more network data might be a double edged sword. On one hand, more information helps to provide a more accurate characterisation of biological phenomenons, and enables some inferences not available in simpler models. On the other hand, it presents confounding factors that may lead to inaccurate conclusions when the underlying meaning of the data is not accounted for.

9.3 Network Modules

The notion of modules is embedded into the conception of network medicine through the *guilt-by-association* principle [10]. While multiple authors [10, 16, 20, 32, 36, 70, 154, 238] have studied the modularity of many biological phenomenons and the genotype-phenotype interplay, they were mostly studied within different contexts. This work portrays different ways in which these different modules connect.

Within the network medicine paradigm, the genotype is defined by a subset of nodes of interest in a biological network [10], while phenotype has a looser definition regarding the object in study. The gene phenotype generally refers to systemic function it serves and it is characterised by the Gene Ontology [3, 121, 139, 148, 238], however it can also refer to the protein interactors as a gain or loss of function [36, 88]. The phenotype can also refer to the observable traits of other processes which occur in an organism, such as disease [20, 26, 90, 205, 207] or drug response [70, 227].

The biological counterpart to the network medicine approach is as follows: different sets of genes are involved in the different cellular functions and response external stimulus, which consist of complex biochemical interactions [12]. These responses can be seen in a cellular level as progression through the cell cycle, changes

in the cytoskeletal structure, cell adhesion, cell migration and cell metabolism, among others, and imply fluctuations in gene expression useful within the intracellular biochemical pathways [152]. Both disease and drug response can be also associated with the up or down regulation of intracellular pathways [76]. Network modules are interesting from a biological point of view since these sets of genes universally appear to cluster in signalling, metabolic, regulatory and PPI networks, and bring a systemic point of view to gene function [212].

However, it is unclear how the modules obtained through the different approaches relate to one another [212]. This work shows that modules of phenotypically similar diseases are coherent with protein function (see Chapter 5), and it continues to validate the usage of network dynamics to prioritise disease gene candidates within the topological module (see Chapter 6). Furthermore, it shows that while drug targets form topological modules, they are heterogeneous and few of them match the corresponding disease module. In particular, Cancer drugs are shown to affect a broad amount of protein functions, as opposed to Bone drugs which target specific functions and affect disease genes.

9.4 Selecting diseases to reduce the noise in Cardigan's prediction

The current understanding of disease mechanisms shows that genetic diseases are very varied. The elucidation of disease's mechanisms of action are often required for the development of detection methods or treatments. At a molecular level, this involves the identification of SNPs and their effects.

Individual or collections of SNPs can cause problems with the gene expression, or affect the protein expressed by the gene. SNPs in a non-coding region may cause under or over expression of a particular gene, both of which can be associated to disease [210]. SNPs in coding regions are associated with changes in the protein structure, which in turn derive into traits associated to disease. Common traits are: loss or gain of function, reduction or gain in activity and interference in activity [217]. It is understandable that at a molecular biology level, the study the mutation trait (e.g. loss-of-function) will benefit from the study of other cases that share the same trait.

Another way to look at the mechanisms of action is from the main phenotypes at an organism level or *pathogenesis*. Some common pathogenesis are inflammation, alterations to the immune response, growth abnormalities, among others. The

research on this area can also link together diseases that share similar phenotypes, and discover biological subsystems involved in those phenotypes [21, 63].

Therefore, it makes sense that a similar approach could be used in computational disease gene prediction models, in particular to avoid noise from possibly confounding mechanisms of action. For instance, the prediction of a mutation on a coding region which derives into loss-of-function, may benefit if the model focuses on diseases which are known to derive from loss-of-function. Then, it is possible for Cardigan to benefit from having a limited amount of diseases in the QWS, to reduce the noise of other phenotypes.

The knowledge that a disease is caused by a genetic loss-of-function is likely to come after the mutant gene is discovered. Therefore, this is not the same stage in which Cardigan is thought to be useful. Cardigan only intends to discover disease-gene associations, so the SNP effect is unlikely to be known at the time of prediction. Notice that if there is a high similarity between the phenotypes of the disease of interest to other diseases that suffer from loss-of-function, it would be reflected in the Caniza similarity. A literature revision of the diseases most similar to the query (measured with the Caniza similarity) is suggested to provide insight on its mechanism of action.

Reducing the set of diseases to focus on a particular phenotype however is not entirely without merit. I believe such a study could be useful, particularly when dealing with a complex disease. Since many genes are expected to cause the disease, it is reasonable to expect that different subsets of genes cause the different traits. For instance if the query disease involves an autoimmune response, it would be interesting to see the top ranked genes while only allowing autoimmune diseases to be in the QWS.

However, only considering particular subsets of diseases to construct the QWS may cause the resulting prediction not to reflect the proper disease module. In principle, Cardigan finds the disease module by approximating modules of other diseases which match every phenotypic characteristic. Recall that the two diseases would be similar according to Caniza similarity if they share any particular phenotype, even if they show other different traits (see 4.3.3). This draws genes that could explain every phenotype shown in the query disease into the Cardigan module. Leaving any of those pieces out may to leave some unexplained phenotypes in the Cardigan prediction. Therefore, I consider that having every possible disease in the QWS is the best solution if a single disease gene must be predicted.

Another way to prioritise Cardigan to match a particular subset of diseases is to retrain the method parameters. This option is certainly a sound strategy to reduce the noise of the prediction. Keeping the previous example, if the researcher is focused on

predicting genes for an autoimmune disease, all (or a subset) of other autoimmune diseases could be used to optimize the diffusion parameters on a *leave-one-out* procedure. This would replace the parameters trained for Acute Lymphoblastic Leukemia in Section 6.1.2. The QWS for the prediction will still contain associations from all the diseases from OMIM, but the balance in the diffusion parameters will be tuned towards characteristics that benefit the discovery of genes for autoimmune diseases.

Notice that the training might take a long time if the training set contains many disease gene associations. A possible drawback of this approach is that the parameters might be overfitted. A cross-validation analysis could be useful to understand the best number of diseases to keep in the training set. Recall that a cross-validation analysis could be very time consuming (as discussed in Section 6.1.2). However,such an analysis could be interesting for future work.

Chapter 10

Future Work

The following projects are continuations of current work or exploratory work that shows promising applications after further development.

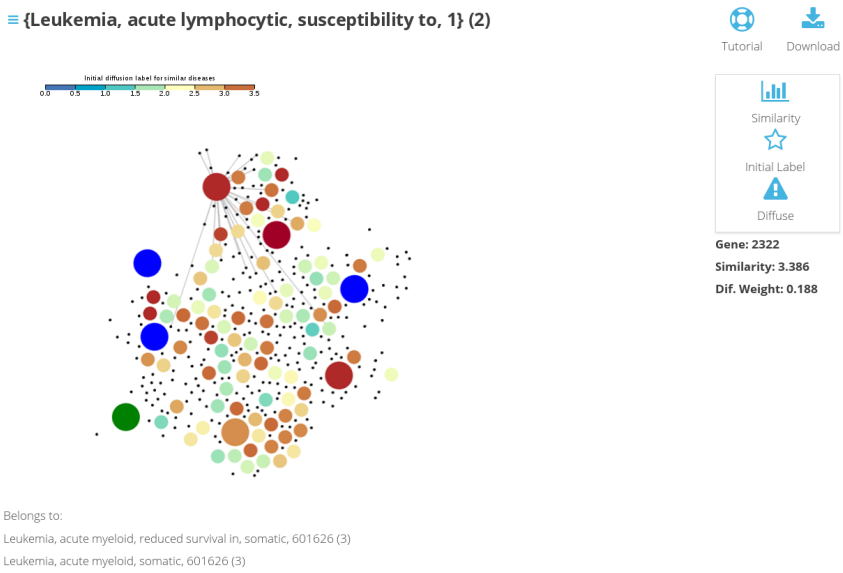
10.1 Cardigan Web

This is a web application intended to be included as a follow up publication for Cardigan. The application is designed contextualise the putative genes given by Cardigan through a network visualisation of the results.

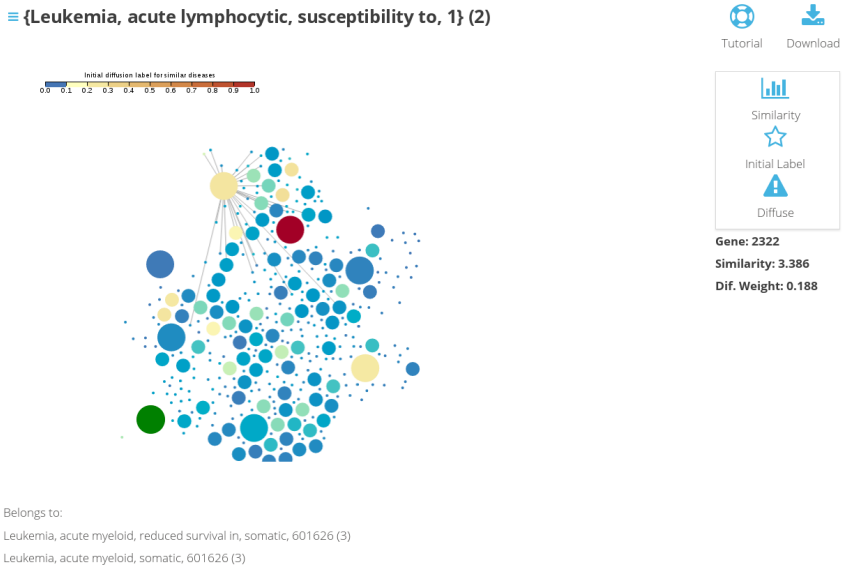
Current features prompt the user to select a query disease, after which Cardigan predicts gene targets for the query which are retrieved to the user (the current implementation calculates the results on the fly, and does not store predictions made for other users). Cardigan Web loads the known disease genes into the *Gene Explorer*, and all their neighbours using the preferred network. The genes in this neighbourhood are highlighted according to the priority assigned by Cardigan.

Notice that this web application is built on HTML5 and JavaScript to produce the visualisation. This allows the server to handle big networks without much of an overload, as the visualisation is rendered entirely on the user's device.

Figure 10.1 shows screenshots from a development stage of the tool, which allows visualisations of how the method changes from known seeds, to the Query Weight Set and to the final prioritisation. The visualisation shows an example of a time-lapse gene prediciton where one gene is the seed found in HPRD for Acute Lymphocytic Leukemia (ALL - MIM:613065), and genes known in 2017 (shown in large circles) are the desired prediction. Notice that while the feature which allows experimental tests resulted interesting during the development of Cardigan, it is likely to be excluded from the final version of Cardigan Web.



(a) Similarity



(b) Final labelling

Fig. 10.1 **Visualisation of a synthetic gene prediction.** The large nodes are the known disease genes and the green one is the gene used as seed for the diffusion process). Genes other than the seed are coloured according to the label in that stage (blue is low and red is high). (a) Shows the genes with the Caniza similarity, and (b) shows the labels after the diffusion process.

10.2 Prediction confidence

This analysis focuses on the possibility of deciding whether it is possible or not to predict which diseases will get a good Cardigan prediction. For instance, Cardigan gets 48% of the targets for *time-lapse* charted diseases within the top 200 results, and 13% of them are predicted in the first position. The problem is to identify which diseases are more likely to be ranked among the top.

The analysis shows that some features based on the semantic similarity of the query disease appear to correlate to the quality of the prediction, and further development appears to be possible. In particular, the generalisation of the analysis appears feasible for methods other than Cardigan, as multiple methods are able to identify some common disease-gene associations among the top predictions.

Analysis of the diffusion input

It is noticeable that not all predictions are equally challenging for the method. In particular, Section 6.2 shows that in average Cardigan has a lower performance on uncharted diseases than on charted diseases. The following proposal considers that the difference Cardigan's performance on different diseases is based on the amount of information used as input for the prediction.

From a machine learning perspective, every instance of disease gene prediction has the same amount of seeds[1], and each disease just changes the initial value of each of those seeds. In fact, the seeds QWS has all the genes known in OMIM: the known genes from the query are set to 1, and genes from other diseases have a value proportional to the similarity of the query (see Section 6.1.1 for the definition of the QWS).

Therefore, each extracted feature is bound to depend on the distribution of values in the QWS representing the query disease. However, the QWS value distribution is not the only important information source, the network connections of the genes associated to the weights are an important orthogonal information source.

Besides the number of known genes $|S|$, two out of several devised features have showed to correlate with the Cardigan's performance: seedness and connectedness. Let $\mathscr{X}(Y^q)$ be Cardigan's performance for disease q using Y^q as the input QWS, then, the Seedness $\mathscr{S}$ is defined as:

$$\mathscr{S}(Y^q) = \sum_{i \in Y^q} y_i,$$

[1]Unless the particular experiment removes some known disease gene associations from the database, such as the leave-one- out predictions.

where y_i is the seed held for gene i in the QWS. The seedness represents the total amount of information known about the disease in q in the QWS. Connectedness $\mathscr{C}$ is defined as:

$$\mathscr{C}(Y^q) = \sum_{(i,j)\in E} y_i y_j W_{ij},$$

where E is the set of edges of the network and W_{ij} is the weight of the edge. The connectedness represents how coherent the QWS information is.

The correlation between the measures ($\mathscr{S}$ *and* $\mathscr{C}$) and the corresponding performances ($\mathscr{X}$) is calculated with a linear regression model. The linear model is used to predict the performances $A = [\mathscr{X}]$ with the measures $B = [\mathscr{S}, \mathscr{C}]$ – i.e. finding β which minimises the RMSE of $A = \beta B$ [14]. Each performance $\mathscr{X}(Y^q)$ is kept in a $[0,1]$ interval by using the AUC of the ROC curve normalised for the first 200 false positives (see Section 4.2.1). Table 10.1 summarises the regression parameters for charted and uncharted diseases.

Table 10.1 **Regression expected gene ranking for seedness and connectedness.** Results are given by leave-one-out tests.

Test set	β_0	$\lvert S\rvert$	$\mathscr{S}$	$\mathscr{C}$
Charted	0.6159	-0.0080	0.0001	-5.623e-05
Uncharted	0.2141	0.1307	4.431e-05	-7.431e-06

Correlation values are not very high, but they suffice to give an idea of the expected performance. The real value corresponding to a predicted performance x can be evaluated by grouping diseases according to ranges of expected performances, and observing the average of the real performance. Table 10.2 shows the results of classifying charted and uncharted diseases in 10 possible bins by their expected performance.

Table 10.2 **Average performances vs expected performances of disease predictions.** The predicted performance of the diseases is used to bin the experiments into brackets. The *Real* column shows average performance of the prediction and the *Expected* column shows the average expected performance for the diseases in the bin.

	Charted		*Uncharted*	
Range	Real	Expected	Real	Expected
0.2 - 0.3	0.316368	0.260080	-	-
0.3 - 0.4	0.386553	0.351363	0.306688	0.375035
0.4 - 0.5	0.397828	0.454885	0.463839	0.473384
0.5 - 0.6	0.597903	0.545644	0.531207	0.525721
0.6 - 0.7	0.625189	0.655033	-	-
0.7 - 0.8	0.748609	0.749621	-	-
0.8 - 0.9	0.807604	0.834941	-	-

While there is variance inside each bin, the real performance matches the expected performance. Nonetheless, it is clear that the regression cannot identify the conditions that grant very low or high performances, and the predictions have a noticeably small range for uncharted diseases. This suggests the validity of the approach and leaves the possibility for further development.

10.3 Integration of heterogeneous networks

The mathematical formulation of the diffusion method for the graph with interfaces (Section 7.1.4) somewhat equates to the unification several diffusion processes that occur on the same protein through different interfaces. Following this intuition, diffusions on several protein networks (e.g. PPI, metabolic, transcriptomic, or others) could be unified considering that the protein will have one interface per network.

The method proposed in Section 7.1.4 (Variant 2) could predict genes in such a network out of the box. However, it is unlikely that each network will bring the same amount of information to the process. The contribution of each network can be controlled with a variation of the interface-interface combination matrix (B_{ij} from Equation 7.5). Another way to control the contributions is to rescale the edge weights of each individual network and give more weight to the most relevant networks.

Other works have used heterogeneous networks for disease gene prediction [50, 54, 107, 223, 231], and can be used as a baseline for the expected gain in performance compared to using only a PPI network.

The problem, objective, tools and basic evaluation procedures seem well defined, making this a good project for further development.

Appendix A

Cardigan Manual

This Appendix serves as a manual for the usage of the Cardigan software. This software is designed as a Python2.7 package, which has been used to produce all the results found in the paper. The software can be downloaded from www.paccanarolab.org/cardigan.

A.1 Quick and simple

Cardigan offers a library to both run the Cardigan algorithm and handle multiple PPI networks, and disease-gene interaction sources (such as OMIM)

Installing

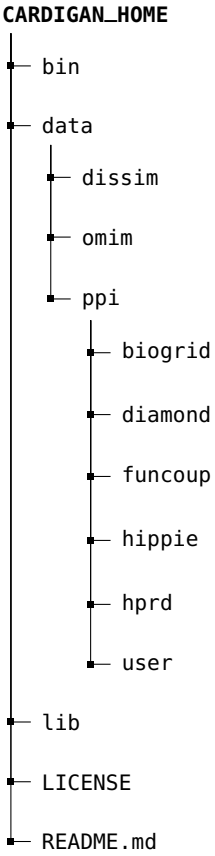
These instructions will show how to setup Cardigan and allow its usage as a Python module.

Prerequisites

Cardigan requires the installation of NumPY and SciPY:

```
$ pip install scipy
```

Furthermore, you need to provide at least one PPI network, a Caniza disease similarity matrix, and both OMIM morbidmap and OMIM mim2gene datasets. The datasets are located by default under the `data` folder in default folder scheme:



The Caniza disease similarity calculated with the 2017 OMIM is provided in the data bundle, otherwise it must be computed by the user. The software to produce the matrix can be downloaded from the paper website.

At least one of the networks (found under `data/ppi`) should be provided. BioGRID in TAB2 format, and DiamondNet from the Ghiassian *et al.* paper (S1 file), FUNCOUP v3.0, HIPPIE and HPRD in their standard format. The custom PPI network should be a tab separated file with Entrez identifiers (called `data/ppi/user/ppi.txt` as default), and may include a third column with the edge weight (all edges should include weights if the PPI is weighted). The default network can be established in the `setPPI(ppiType)` method found `lib/Config.py` (it defaults to `'custom'`).

All file names and locations can be changed in `lib/Config.py`.

Library setup

The modules can run as they are, however, the Caniza similarity must be pre-processed for faster access within the method. The pre-process and data availability can be done launching the setup:

```
$ cd CARDIGAN_HOME/bin
$ python setup.py
```

If the setup is not executed, the Caniza similarity will be pre-processed the first time Cardigan runs.

Running Cardigan

You can find working examples under the `bin` folder.

The following example is the default use case of the library. Further details are provided by the *Cookbook* Section, which shows examples to run the experiments from this work.

Minimum working example

This is the basic usage of the software

```
import sys
sys.path.append('../lib')

import Cardigan

# load the module using the default configuration
cardigan = Cardigan.Cardigan()

# predict genes for BDPLT16 (MIM: 187800)
out = cardigan.predict(['187800'])
print out
```

A.2 Cookbook

This section covers the most frequent usage of Cardigan with short and simple examples. In general these recipes can be combined according to the user's need.

Customise the known genes

If the seeds are defined all known genes for the disease are replaced with the provided seeds

```
import sys
sys.path.append('../lib')

import Cardigan

# load the module using the default configuration
cardigan = Cardigan.Cardigan()

# predict genes for BDPLT16 (MIM: 187800)
# set gene 3690 as a seed
out = cardigan.predict(['187800'],['3690'])

```

```
13 print out
```

Customise the targets and evaluate

Cardigan offers some evaluation measures to assess the predictions. Providing a target allows Cardigan the ability to evaluate the last prediction. Importantly, the target cannot be one of the known seeds of the disease being predicted.

This example defines seeds and targets to predict a synthetically removed disease gene

```
1  import sys
2  sys.path.append('../lib')
3
4  import Cardigan
5
6  # load the module using the default configuration
7  cardigan = Cardigan.Cardigan()
8
9  # predict genes for BDPLT16 (MIM: 187800)
10 out = cardigan.predict(['187800'],seedGenes=['3690'],targetGenes=['3674'])
11 targetPos = cardigan.evaluate()
12
13 print targetPos
```

Changing the PPI network

You can establish the network by loading the appropriate `config` to Cardigan:

```
1  import sys
2  sys.path.append('../lib')
3
4
5  import Cardigan
6  import Config
7
8  # load the configuration module
9  config = Config.Config()
10 config.setPPI('hprd')
11 # load the module with the modified configuration
12 cardigan = Cardigan.Cardigan(config)
13
14 # predict genes for BDPLT16 (MIM: 187800)
15 # keep gene 3690 as a seed and use 3674 as a target
16 out = cardigan.predict(['187800'],['3690'],['3674'])
17 targetPos = cardigan.evaluate()
18 print targetPos
```

The supported networks are `'hprd'`, `'diamondnet'`, `'biogrid'`, `'hippie'`, `'funcoup'`, and `'custom'` (which is explained in A.3.1).

Experimental setups

In addition to the prediction method, Cardigan offers further utilities to handle the gene associations and produce different experimental setups.

Time-lapse experiments

These experiments require both old and new disease gene association files

```
import sys
sys.path.append('../lib')

import Config
import OMIMUtilities

# load the configuration file
config = Config.Config()

# OMIM provides a gene MIM to Entrez mapping, which is loaded here
mim2gene = OMIMUtilities.Mim2Gene(config.mim2gene)
# then the disease gene association mapping is loaded
# for the old OMIM
mmo =  OMIMUtilities.MorbidMap(config.morbidMapOld,
  separator=config.morbidMapOldSep, mim2gene=mim2gene)
# and for the new OMIM
mmn =  OMIMUtilities.MorbidMap(config.morbidMapNew, separator=config.
    morbidMapNewSep, mim2gene=mim2gene, oldMorbidmap=mmo)

print 'Diseases with more genes', len(mmn.geneAdditions)
print 'Previously uncharted diseases', len(mmn.diseaseAdditions)
```

Then, the experiment can be produced in a loop

```
import Cardigan

# load the module using the default configuration
cardigan = Cardigan.Cardigan()

# for each charted disease that gained some more
for disease in mmn.geneAdditions:
  print 'Disease',disease

  # get one target at a time
  for target in mmn.geneAdditions[disease]:

    # predict using the seeds from the old database
    out = cardigan.predict(disease, mmo.getGenes(disease), target)

    # evaluate the position of the target
    targetPos = cardigan.evaluate()
    print target, 'found in position', targetPos
```

Leave-one-out experiments

These experiments require a single gene association file

```
import sys
sys.path.append('../lib')

import Config
import OMIMUtilities

# load the configuration file
config = Config.Config()

# OMIM provides a gene MIM to Entrez mapping, which is loaded here
mim2gene = OMIMUtilities.Mim2Gene(config.mim2gene)
# then the disease gene association mapping is loaded
morbidMap =  OMIMUtilities.MorbidMap(config.morbidMap, separator=config.
    morbidMapSep, mim2gene=mim2gene)

# obtain the genes associated to BDPLT16 (MIM: 187800)
disease = ['187800']
knownGenes = morbidMap.getGenes(disease)

print knownGenes
```

Then, the *leave-one-out* procedure is a simple loop

```
import Cardigan

# load the module using the default configuration
cardigan = Cardigan.Cardigan()

# get one target at a time
for target in knownGenes:
  # all other known genes are set as seeds
  # i.e. one gene was left out
  seeds = [x for x in knownGenes if x != target]

  # predict using the left over seeds
  out = cardigan.predict(disease, seeds, target)

  # evaluate the position of the target
  targetPos = cardigan.evaluate()
  print target, 'found in position', targetPos
```

A.3 Configuration

A.3.1 Network handlers

The network handlers are objects which parse text-based protein-protein interaction networks and provide an interface for Cardigan.

Expected use

The user can provide a custom PPI network in a TSV format, which is located by default in `data/ppi/user/ppi.txt`.

```
# lines starting with # are comments
# Entrez_Protein_A__Entrez_Protein_B____Weight
entrezA___entrezB___WeightAB
entrezA___entrezC___WeightAC
entrezB___entrezD___WeightBD
```

After providing the interaction file, set the custom PPI in the configuration

```
import sys
sys.path.append('../lib')


import Cardigan
import Config

# load the configuration module
config = Config.Config()
config.setPPI('custom')
# load the module with the modified configuration
cardigan = Cardigan.Cardigan(config)

# predict genes for BDPLT16 (MIM: 187800)
out = cardigan.predict(['187800'])
print out
```

Advanced use

The user can directly produce a PPI object and override the configuration object for its usage within Cardigan. The modifications follow the structure:

```
import sys
sys.path.append('../lib')

import PPIUtilities
import Config
import Cardigan

# load the configuration file
config = Config.Config()

# Load the PPI object
ppi = PPIUtilities.PPI('../data/ppi/user/myCustomPPI.tsv', **myCustomParameters)

# update the configuration with the custom PPI
config.ppiObject = ppi
# load the module with the modified configuration
cardigan = Cardigan.Cardigan(config)

# predict genes for BDPLT16 (MIM: 187800)
out = cardigan.predict(['187800'])
print out
```

All intended functionality is provider from the object constructor

```
PPIUtilities.PPI(filename, columns, ppiName, mainIdentifier,
  separator, synSeparator, translateFile, translateCols,
  verbose, simplify)
```

- `filename` is the only mandatory field and should reference a text file.
- `columns` is a list of column identifiers which should match the number of columns from the file. E. g. `['entrez_a','entrez_b','weight']`, where the endings `'_a'` and `'_b'` are not optional.
- `ppiName` is a string to identify the object.
- `mainIdentifier` is a string representing the column identifiers to be considered for the network. E. g. `'entrez'`.
- `separator` is the column separator (i.e. `'\t'` for a tab separated file).
- `synSeparator` is the character user to separate gene symbols. Gene symbols (identified by columns `'symbol_a'` and `'symbol_b'`) are used in the lines where the main identifier is missing.
- `translateFile` is a reference to an additional 2 column tab separated file used to translate the main identifier. It is used for instance if the custom dataset contains UNIPROT identifiers that must be translated to Entrez.
- `translateCols` is the list `['source','target']` or `['target','source']`, which specifies the order of the columns in the translate file.
- `verbose` is a boolean where **`False`** prevents the summary of the network mapping to be printed in the console,**`True`** is the default otherwise.
- `simplify` is a boolean where **`False`** indicates that self loops are allowed, **`True`** is the default otherwise.

Appendix B

Data mapping

Auxiliary data mappings crafted for different projects are presented here.

B.1 Diseases to OMIM

The following table presents a translation between disease names and OMIM identifiers. The disease names are defined by Ghiassian *et al.* [55] and serve roughly as disease families which were used for several disease gene prediction methods [90, 139, 148]. The OMIM identifiers are obtained by collecting diseases with similar names or by matching the description.

Notice that the OMIM identifiers provided here are used as the query for Cardigan. While each individual disease from the family contains known gene associations within OMIM, all those genes are synthetically removed from OMIM for the construction of the Query Weight Set. Instead, the disease family is associated to the genes provided by the Ghiassian mapping [55].

Table B.1 **Ghiassian to OMIM identifier translation.** Each disease name is associated to a set of OMIM identifiers, which is considered to define the disease (family).

No	*Disease*	*OMIM*
1	adrenal gland diseases	219080, 300200, 613677, 202010, 201710, 201910, 201810, 300018
2	alzheimer disease	104300, 607822, 104310, 606889

continues on next page

No	*Disease*	*OMIM*
3	amino acid metabolism inborn errors	222700, 609924, 608643, 231670, 220100
4	amyotrophic lateral sclerosis	105400, 205100, 611895, 606070, 608627, 602433, 602099, 614696
5	anemia aplastic	609135
6	anemia hemolytic	205700, 613470, 266120, 612631, 300908, 235700, 612300, 231900, 102730, 230450
7	aneurysm	607086, 105805
8	arrhythmias cardiac	115000
9	arterial occlusive diseases	602531
10	arteriosclerosis	208060
11	arthritis rheumatoid	180300, 604302
12	asthma	600807
13	basal ganglia diseases	213600, 606656, 615007, 615483, 616413, 114100
14	behcet syndrome	109650
15	bile duct diseases	603003, 608063
16	blood coagulation disorders	306700, 306900, 227500, 227600, 612416
17	blood platelet disorders	139090, 601399, 601709, 609821, 605249, 231200
18	breast neoplasms	114480
19	carbohydrate metabolism inborn errors	212066, 212065
20	carcinoma renal cell	144700

continues on next page

No	*Disease*	*OMIM*
21	cardiomyopathies	500000, 115210
22	cardiomyopathy hypertrophic	115200, 192600
23	celiac disease	212750
24	cerebellar ataxia	224050, 604121
25	cerebrovascular disorders	182410, 601367, 301500
26	charcot-marie-tooth disease	118220, 118200
27	colitis ulcerative	266600
28	colorectal neoplasms	114500
29	coronary artery disease	608320, 168820
30	crohn disease	266600
31	death sudden	272120
32	diabetes mellitus type 2	125853
33	dwarfism	210710, 168400
34	esophageal diseases	614266, 164280, 109350
35	exophthalmos	605039, 259775, 170100
36	glomerulonephritis	305800, 248760
37	gout	300661
38	graves disease	275000, 300351
39	head and neck neoplasms	275355, 162200
40	hypothalamic diseases	262400, 143100, 275120, 176270, 614962
41	leukemia b-cell	613065
42	leukemia myeloid	601626
43	lipid metabolism disorders	275630, 602579, 212065, 107730

continues on next page

No	*Disease*	*OMIM*
44	liver cirrhosis	215600
45	liver cirrhosis biliary	109720, 613007, 613008, 614220, 614221
46	lung diseases obstructive	606963
47	lupus erythematosus	152700, 607279
48	lymphoma	605027, 151430, 186960
49	lysosomal storage diseases	300257, 256540, 232300, 248500, 230500
50	macular degeneration	603075, 607921
51	metabolic syndrome x	605552
52	motor neuron disease	600333, 105550
53	multiple sclerosis	126200, 612594, 612595, 612596, 614810
54	muscular dystrophies	300377, 158900, 310200
55	mycobacterium infections	607948, 209950
56	myeloproliferative disorders	613523, 131440
57	nutritional and metabolic diseases	277400, 603358
58	peroxisomal disorders	214100, 601539
59	psoriasis	177900, 602723
60	purine-pyrimidine metabolism inborn errors	613161, 274270
61	renal tubular transport inborn errors	602722, 611590, 179800
62	sarcoma	300813, 259500, 117600
63	spastic paraplegia hereditary	182600
64	spinocerebellar ataxias	164400

continues on next page

No	*Disease*	*OMIM*
65	spinocerebellar degenerations	183090
66	spondylarthropathies	106300
67	tauopathies	600274
68	uveal diseases	155720, 120433
69	varicose veins	192200
70	vasculitis	192310, 609817

B.2 DrugBank Categories

The following mapping associates Disease categories to Drug categories. Drugs from a given category are expected to be used for the treatment of diseases from the corresponding category. This mapping covers about 60% of the FDA approved drugs found in DrugBank 5.05 [219].

The Goh *et al.* categories [57] are used to define the disease categories. The corresponding drug categories are Anatomical Therapeutic Chemical (ATC) categories [146] when possible, or DrugBank categories [219] on defect. The complete mapping is presented in Table B.2.

Table B.2 **Disease category to Drug category mapping.** Disease categories are based on the Goh *et al.* classification and drug categories are provided ATC when possible or fallback to DrugBank categories.

Disease Category	*Drug Category*	
Goh	ATC	DrugBank
Bone	-	DBCAT000089, DBCAT002165
Cancer	L01	-
Cardiovascular	C, V09G	-
Connective tissue	V03AK	-
Dermatological	D	-
Developmental	H01	-

continues on next page

Disease Category	*Drug Category*	
Goh	ATC	DrugBank
Ear-Nose-Throat	S02	DBCAT002159, DBCAT002106
Endocrine	L02	-
Gastrointestinal	A03	-
Hematological	B	-
Immunological	L03, L04	-
Metabolic	A09, A14, A15, A16	-
Muscular, Skeletal	M	-
Neurological	N	-
Nutritional	-	DBCAT000234, DBCAT002733
Ophthalmologic	S01	-
Psychiatric	-	DBCAT000529
Renal	-	DBCAT000629
Respiratory	R	-

Notice that the *Multiple* Goh category is left out of the mapping, as each disease requires an independent drug association. The *Muscular* and *Skeletal* Goh categories are matched to the DrugBank *Musculoskeletal* category (M), to avoid inaccurate separations of the musculoskeletal drugs. A summary of the FDA approved drugs, drug targets and disease proteins per category is shown in Table B.3. Over 60% of FDA approved drugs are mapped to a Goh category. This provides a good coverage of the drug set, and a mapping extension may reduce the confidence in the associations of drugs to Goh categories without expert curation.

Table B.3 **Summary of the Goh to DrugBank category mapping**. The middle and right column show the number of drugs and diseases per category. Notice that the total is not the direct sum of the numbers as some diseases and drugs belong to multiple categories.

Category	*Diseases*	*Drugs*
Bone	52	25
Cancer	198	138
Cardiovascular	98	223
Connective tissue	49	0

continues on next page

Category	*Diseases*	*Drugs*
Dermatological	95	140
Developmental	51	21
Ear-Nose-Throat	52	62
Endocrine	99	22
Gastrointestinal	34	25
Hematological	157	96
Immunological	105	58
Metabolic	291	32
Musculoskeletal	167	91
Neurological	270	302
Nutritional	12	47
Ophthalmologic	149	137
Psychiatric	26	47
Renal	59	14
Respiratory	22	145
Overall	1928	1302

References

[1] Abruzzo, M. and Erickson, R. (1977). A new syndrome of cleft palate associated with coloboma, hypospadias, deafness, short stature, and radial synostosis. *Journal of medical genetics*, 14(1):76–80.

[2] Adamcsek, B., Palla, G., Farkas, I. J., Derényi, I., and Vicsek, T. (2006). Cfinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, 22(8):1021–1023.

[3] Aerts, S., Lambrechts, D., Maity, S., Van Loo, P., Coessens, B., De Smet, F., Tranchevent, L.-C., De Moor, B., Marynen, P., Hassan, B., et al. (2006). Gene prioritization through genomic data fusion. *Nature biotechnology*, 24(5):537–544.

[4] Aerts, S., Van Loo, P., Moreau, Y., and De Moor, B. (2004). A genetic algorithm for the detection of new cis-regulatory modules in sets of coregulated genes. *Bioinformatics*, 20(12):1974–1976.

[5] Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47.

[6] Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410.

[7] Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., et al. (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29.

[8] Bader, G. D., Betel, D., and Hogue, C. W. (2003). Bind: the biomolecular interaction network database. *Nucleic acids research*, 31(1):248–250.

[9] Barabasi, A.-L., Gulbahce, N., and Loscalzo, J. (2011). Network medicine: a network-based approach to human disease. *Nature Reviews Genetics*, 12(1):56–68.

[10] Barabasi, A.-L. and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101.

[11] Barnes, A. M., Duncan, G., Weis, M., Paton, W., Cabral, W. A., Mertz, E. L., Makareeva, E., Gambello, M. J., Lacbawan, F. L., Leikin, S., et al. (2013). Kuskokwim syndrome, a recessive congenital contracture disorder, extends the phenotype of fkbp10 mutations. *Human mutation*, 34(9):1279–1288.

[12] Basbaum, A. I., Bautista, D. M., Scherrer, G., and Julius, D. (2009). Cellular and molecular mechanisms of pain. *Cell*, 139(2):267–284.

[13] Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L., et al. (2004). The pfam protein families database. *Nucleic acids research*, 32(suppl_1):D138–D141.

[14] Bishop, C. M. (2006). Pattern recognition. *Machine Learning*.

[15] Brenner, R., Vetter, U., Stöss, H., Müller, P., and Teller, W. (1993). Defective collagen fibril formation and mineralization in osteogenesis imperfecta with congenital joint contractures (bruck syndrome). *European journal of pediatrics*, 152(6):505–508.

[16] Brohee, S. and Van Helden, J. (2006). Evaluation of clustering algorithms for protein-protein interaction networks. *BMC bioinformatics*, 7(1):488.

[17] Cáceres, J. J. and Paccanaro, A. (2016). Combining interactomes from multiple organisms: A case study on human-mouse. In *Computing Conference (CLEI), 2016 XLII Latin American*, pages 1–6. IEEE.

[18] Cáceres, J. J. and Paccanaro, A. (2017). An exploratory analysis on drug target locality. In *Simposio Argentino de GRANdes DAtos (AGRANDA)-JAIIO 46 (Córdoba, 2017)*.

[19] Caniza, H., Romero, A. E., Heron, S., Yang, H., Devoto, A., Frasca, M., Mesiti, M., Valentini, G., and Paccanaro, A. (2014). Gossto: a stand-alone application and a web tool for calculating semantic similarities on the gene ontology. *Bioinformatics*, 30(15):2235–2236.

[20] Caniza, H., Romero, A. E., and Paccanaro, A. (2015). A network medicine approach to quantify distance between hereditary disease modules on the interactome. *Scientific reports*, 5.

[21] Casqueiro, J., Casqueiro, J., and Alves, C. (2012). Infections in patients with diabetes mellitus: A review of pathogenesis. *Indian journal of endocrinology and metabolism*, 16(Suppl1):S27.

[22] Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15.

[23] Chapelle, O., Scholkopf, B., and Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542.

[24] Chatr-Aryamontri, A., Breitkreutz, B.-J., Oughtred, R., Boucher, L., Heinicke, S., Chen, D., Stark, C., Breitkreutz, A., Kolas, N., O'Donnell, L., et al. (2015). The BioGRID interaction database: 2015 update. *Nucleic acids research*, 43(D1):D470–D478.

[25] Chen, J., Aronow, B. J., and Jegga, A. G. (2009). Disease candidate gene identification and prioritization using protein interaction networks. *BMC bioinformatics*, 10(1):1.

[26] Chen, R., Mias, G. I., Li-Pook-Than, J., Jiang, L., Lam, H. Y., Chen, R., Miriami, E., Karczewski, K. J., Hariharan, M., Dewey, F. E., et al. (2012). Personal omics profiling reveals dynamic molecular and medical phenotypes. *Cell*, 148(6):1293–1307.

[27] Chung, F. R. and Graham, F. C. (1997). *Spectral graph theory*. Number 92 in Regional Conference Series in Mathematics. American Mathematical Soc.

[28] Collins, S. R., Kemmeren, P., Zhao, X.-C., Greenblatt, J. F., Spencer, F., Holstege, F. C., Weissman, J. S., and Krogan, N. J. (2007). Toward a comprehensive atlas of the physical interactome of Saccharomyces cerevisiae. *Mol Cell Proteomics*, 6:439–450.

[29] Consortium, . G. P. et al. (2012). An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56.

[30] Consortium, I. H. G. S. et al. (2001). Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860.

[31] Consortium, U. et al. (2018). Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 46(5):2699.

[32] Couto, F. M. and Silva, M. J. (2011). Disjunctive shared information between ontology concepts: application to gene ontology. *Journal of biomedical semantics*, 2(1):5.

[33] Croft, D., O'Kelly, G., Wu, G., Haw, R., Gillespie, M., Matthews, L., Caudy, M., Garapati, P., Gopinath, G., Jassal, B., et al. (2010). Reactome: a database of reactions, pathways and biological processes. *Nucleic acids research*, page gkq1018.

[34] Dagley, S. and CHAPMAN, P. (1971). Metabolic pathways. *Methods in Microbiology*, page 217.

[35] Dale, J. M., Popescu, L., and Karp, P. D. (2010). Machine learning methods for metabolic pathway prediction. *BMC bioinformatics*, 11(1):15.

[36] Das, J., Fragoza, R., Lee, H. R., Cordero, N. A., Guo, Y., Meyer, M. J., Vo, T. V., Wang, X., and Yu, H. (2014). Exploring mechanisms of human disease through structurally resolved protein interactome networks. *Molecular BioSystems*, 10(1):9–17.

[37] Davey Smith, G. and Ebrahim, S. (2003). 'Mendelian randomization': can genetic epidemiology contribute to understanding environmental determinants of disease? *International journal of epidemiology*, 32(1):1–22.

[38] de Vries, S. J. and Bonvin, A. M. (2011). Cport: a consensus interface predictor and its performance in prediction-driven docking with haddock. *PloS one*, 6(3):e17695.

[39] Desmedt, C., Haibe-Kains, B., Wirapati, P., Buyse, M., Larsimont, D., Bontempi, G., Delorenzi, M., Piccart, M., and Sotiriou, C. (2008). Biological processes associated with breast cancer clinical outcome depend on the molecular subtypes. *Clinical cancer research*, 14(16):5158–5165.

[40] di Bernardo, D., Thompson, M. J., Gardner, T. S., Chobot, S. E., Eastwood, E. L., Wojtovich, A. P., Elliott, S. J., Schaus, S. E., and Collins, J. J. (2005). Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature biotechnology*, 23(3):377.

[41] Djebali, S., Davis, C. A., Merkel, A., Dobin, A., Lassmann, T., Mortazavi, A., Tanzer, A., Lagarde, J., Lin, W., Schlesinger, F., et al. (2012). Landscape of transcription in human cells. *Nature*, 489(7414):101.

[42] Edwards, A., Kus, B., Jansen, R., Greenbaum, D., Greenblatt, J., and Gerstein, M. (2002). Bridging structural biology and genomics: assessing protein interaction data with known complexes. *Trends Genet*, 18(10):529–36.

[43] Edwards, Y. J., Beecham, G. W., Scott, W. K., Khuri, S., Bademci, G., Tekin, D., Martin, E. R., Jiang, Z., Mash, D. C., Pericak-Vance, M. A., et al. (2011). Identifying consensus disease pathways in parkinson's disease using an integrative systems biology approach. *PLoS one*, 6(2):e16917.

[44] Emilsson, V., Thorleifsson, G., Zhang, B., Leonardson, A. S., Zink, F., Zhu, J., Carlson, S., Helgason, A., Walters, G. B., Gunnarsdottir, S., et al. (2008). Genetics of gene expression and its effect on disease. *Nature*, 452(7186):423.

[45] Eppig, J. T., Blake, J. A., Bult, C. J., Kadin, J. A., Richardson, J. E., and Group, M. G. D. (2014). The mouse genome database (mgd): facilitating mouse as a model for human biology and disease. *Nucleic acids research*, 43(D1):D726–D736.

[46] Evans, W. E. and McLeod, H. L. (2003). Pharmacogenomics—drug disposition, drug targets, and side effects. *New England Journal of Medicine*, 348(6):538–549.

[47] Fick, A. (1855). V. on liquid diffusion. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 10(63):30–39.

[48] Fisher, R. A. (1919). Xv.—the correlation between relatives on the supposition of mendelian inheritance. *Earth and Environmental Science Transactions of the Royal Society of Edinburgh*, 52(2):399–433.

[49] Forbes, S. A., Beare, D., Boutselakis, H., Bamford, S., Bindal, N., Tate, J., Cole, C. G., Ward, S., Dawson, E., Ponting, L., et al. (2016). Cosmic: somatic cancer genetics at high-resolution. *Nucleic acids research*, 45(D1):D777–D783.

[50] Franke, L., Van Bakel, H., Fokkens, L., De Jong, E. D., Egmont-Petersen, M., and Wijmenga, C. (2006). Reconstruction of a functional human gene network, with an application for prioritizing positional candidate genes. *The American Journal of Human Genetics*, 78(6):1011–1025.

[51] Fromer, M., Roussos, P., Sieberts, S. K., Johnson, J. S., Kavanagh, D. H., Perumal, T. M., Ruderfer, D. M., Oh, E. C., Topol, A., Shah, H. R., et al. (2016). Gene expression elucidates functional impact of polygenic risk for schizophrenia. *Nature neuroscience*, 19(11):1442.

[52] Futreal, P. A., Coin, L., Marshall, M., Down, T., Hubbard, T., Wooster, R., Rahman, N., and Stratton, M. R. (2004). A census of human cancer genes. *Nature reviews cancer*, 4(3):177.

[53] Gammerman, A., Vovk, V., and Vapnik, V. (1998). Learning by transduction. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 148–155. Morgan Kaufmann Publishers Inc.

[54] George, R. A., Liu, J. Y., Feng, L. L., Bryson-Richardson, R. J., Fatkin, D., and Wouters, M. A. (2006). Analysis of protein sequence and interaction data for candidate disease gene prediction. *Nucleic acids research*, 34(19):e130–e130.

[55] Ghiassian, S. D., Menche, J., and Barabási, A.-L. (2015). A disease module detection (diamond) algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the human interactome. *PLoS computational biology*, 11(4):e1004120.

[56] Glass, I., Swindlehurst, C., Aitken, D., McCrea, W., and Boyd, E. (1989). Interstitial deletion of the long arm of chromosome 2 with normal levels of isocitrate dehydrogenase. *Journal of medical genetics*, 26(2):127–130.

[57] Goh, K.-I., Cusick, M. E., Valle, D., Childs, B., Vidal, M., and Barabási, A.-L. (2007). The human disease network. *Proceedings of the National Academy of Sciences*, 104(21):8685–8690.

[58] Gottlieb, A., Magger, O., Berman, I., Ruppin, E., and Sharan, R. (2011a). Principle: a tool for associating genes with diseases via network propagation. *Bioinformatics*, 27(23):3325–3326.

[59] Gottlieb, A., Stein, G. Y., Ruppin, E., and Sharan, R. (2011b). Predict: a method for inferring novel drug indications with application to personalized medicine. *Molecular systems biology*, 7(1):496.

[60] Gross, J. L. and Yellen, J. (2005). *Graph theory and its applications*. Chapman and Hall/CRC.

[61] Guénet, J. L. (2005). The mouse genome. *Genome research*, 15(12):1729–1740.

[62] Han, J.-D. J., Bertin, N., Hao, T., Goldberg, D. S., Berriz, G. F., Zhang, L. V., Dupuy, D., Walhout, A. J., Cusick, M. E., Roth, F. P., et al. (2004). Evidence for dynamically organized modularity in the yeast protein–protein interaction network. *Nature*, 430(6995):88.

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

[64] Hart, G. T., Lee, I., and Marcotte, E. M. (2007). A high-accuracy consensus map of yeast protein complexes reveals modular nature of gene essentiality. *BMC Bioinformatics*, 8:236.

[65] Hill, S. M., Heiser, L. M., Cokelaer, T., Unger, M., Nesser, N. K., Carlin, D. E., Zhang, Y., Sokolov, A., Paull, E. O., Wong, C. K., et al. (2016). Inferring causal molecular networks: empirical assessment through a community-based effort. *Nature methods*, 13(4):310.

[66] Hindorff, L. A., Sethupathy, P., Junkins, H. A., Ramos, E. M., Mehta, J. P., Collins, F. S., and Manolio, T. A. (2009). Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proceedings of the National Academy of Sciences*, 106(23):9362–9367.

[67] Ho, Y., Gruhler, A., Heilbut, A., Bader, G., Moore, L., Adams, S., Millar, A., Taylor, P., Bennett, K., Boutilier, K., Yang, L., Wolting, C., Donaldson, I., Schandorff, S., Shewnarane, J., Vo, M., Taggart, J., Goudreault, M., Muskat, B., Alfarano, C., Dewar, D., Lin, Z., Michalickova, K., Willems, A., Sassi, H., Nielsen, P., Rasmussen, K., Andersen, J., Johansen, L., Hansen, L., Jespersen, H., Podtelejnikov, A., Nielsen, E., Crawford, J., Poulsen, V., Sørensen, B., Matthiesen, J., Hendrickson, R., Gleeson, F., Pawson, T., Moran, M., Durocher, D., Mann, M., Hogue, C., Figeys, D., and Tyers, M. (2002). Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature*, 415(6868):180–3.

[68] Hofmann, T., Schölkopf, B., and Smola, A. J. (2008). Kernel methods in machine learning. *The annals of statistics*, pages 1171–1220.

[69] Home Reference, G. (2018). *Help Me Understand Genetics Genomic Research*. Lister Hill National Center for Biomedical Communications. U.S. National Library of Medicine. National Institutes of Health. Department of Health & Human Services. https://ghr.nlm.nih.gov/.

[70] Hopkins, A. L. (2007). Network pharmacology. *Nature biotechnology*, 25(10):1110.

[71] Hwang, H., Vreven, T., and Weng, Z. (2014). Binding interface prediction by combining protein–protein docking results. *Proteins: Structure, Function, and Bioinformatics*, 82(1):57–66.

[72] Imakaev, M., Fudenberg, G., McCord, R. P., Naumova, N., Goloborodko, A., Lajoie, B. R., Dekker, J., and Mirny, L. A. (2012). Iterative correction of hi-c data reveals hallmarks of chromosome organization. *Nature methods*, 9(10):999.

[73] Imming, P., Sinning, C., and Meyer, A. (2006). Drugs, their targets and the nature and number of drug targets. *Nature reviews Drug discovery*, 5(10):821.

[74] Iorio, F., Saez-Rodriguez, J., and Di Bernardo, D. (2013). Network based elucidation of drug response: from modulators to targets. *BMC systems biology*, 7(1):139.

[75] Iskar, M., Campillos, M., Kuhn, M., Jensen, L. J., Van Noort, V., and Bork, P. (2010). Drug-induced regulation of target expression. *PLoS computational biology*, 6(9):e1000925.

[76] Jenkins, J. L. (2013). Drug discovery: Rethinking cellular drug response. *Nature chemical biology*, 9(11):669.

[77] Jia, P. and Zhao, Z. (2014). Network-assisted analysis to prioritize gwas results: principles, methods and perspectives. *Human genetics*, 133(2):125–138.

[78] Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.

[79] Jiang, Y., Oron, T. R., Clark, W. T., Bankapur, A. R., D'Andrea, D., Lepore, R., Funk, C. S., Kahanda, I., Verspoor, K. M., Ben-Hur, A., et al. (2016). An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome biology*, 17(1):184.

[80] Jin, G., Fu, C., Zhao, H., Cui, K., Chang, J., and Wong, S. T. (2011). A novel method of transcriptional response analysis to facilitate drug repositioning for cancer therapy. *Cancer research*.

[81] Jordan, R. A., Yasser, E.-M., Dobbs, D., and Honavar, V. (2012). Predicting protein-protein interface residues using local surface structural similarity. *BMC bioinformatics*, 13(1):41.

[82] Kanehisa, M. and Goto, S. (2000). Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30.

[83] Karni, S., Soreq, H., and Sharan, R. (2009). A network-based method for predicting disease-causing genes. *Journal of Computational Biology*, 16(2):181–189.

[84] Karpe, P. D., Latendresse, M., and Caspi, R. (2011). The pathway tools pathway prediction algorithm. *Standards in genomic sciences*, 5(3):424.

[85] Kastritis, P. L. and Bonvin, A. M. (2013). On the binding affinity of macromolecular interactions: daring to ask why proteins interact. *Journal of The Royal Society Interface*, 10(79):20120835.

[86] Kawamoto, K., Houlihan, C. A., Balas, E. A., and Lobach, D. F. (2005). Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. *Bmj*, 330(7494):765.

[87] Khatri, P. and Drăghici, S. (2005). Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*, 21(18):3587–3595.

[88] Khurana, E., Fu, Y., Colonna, V., Mu, X. J., Kang, H. M., Lappalainen, T., Sboner, A., Lochovsky, L., Chen, J., Harmanci, A., et al. (2013). Integrative annotation of variants from 1092 humans: application to cancer genomics. *Science*, 342(6154):1235587.

[89] King, A. D., Pržulj, N., and Jurisica, I. (2004). Protein complex prediction via cost-based clustering. *Bioinformatics*, 20(17):3013–3020.

[90] Köhler, S., Bauer, S., Horn, D., and Robinson, P. N. (2008). Walking the interactome for prioritization of candidate disease genes. *The American Journal of Human Genetics*, 82(4):949–958.

[91] Köhler, S., Doelken, S. C., Mungall, C. J., Bauer, S., Firth, H. V., Bailleul-Forestier, I., Black, G. C., Brown, D. L., Brudno, M., Campbell, J., et al. (2013). The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research*, 42(D1):D966–D974.

[92] Kondor, R. I. and Lafferty, J. (2002). Diffusion kernels on graphs and other discrete structures. In *Proceedings of the 19th international conference on machine learning*, volume 2002, pages 315–322.

[93] Kortemme, T., Kim, D. E., and Baker, D. (2004). Computational alanine scanning of protein-protein interfaces. *Sci. STKE*, 2004(219):pl2–pl2.

[94] Kraus, J. M., Tatipaka, H. B., McGuffin, S. A., Chennamaneni, N. K., Karimi, M., Arif, J., Verlinde, C. L., Buckner, F. S., and Gelb, M. H. (2010). Second generation analogues of the cancer drug clinical candidate tipifarnib for anti-chagas disease drug discovery. *Journal of medicinal chemistry*, 53(10):3887–3898.

[95] Krogan, N., Cagney, G., Yu, H., Zhong, G., Guo, X., Ignatchenko, A., Li, J., Pu, S., Datta, N., Tikuisis, A., Punna, T., Peregrin-Alvarez, J., Shales, M., Zhang, X., Davey, M., Robinson, M., Paccanaro, A., Bray, J., Sheung, A., Beattie, B., Richards, D., Canadien, V., Lalev, A., Mena, F., Wong, P., Starostine, A., Canete, M., Vlasblom, J., Wu, S., Orsi, C., Collins, S., Chandran, S., Haw, R., Rilstone, J., Gandi, K., Thompson, N., Musso, G., St Onge, P., Ghanny, S., Lam, M., Butland, G., Altaf-Ui, A., Kanaya, S., Shilatifard, A., O'Shea, E., Weissman, J., Ingles, C., Hughes, T., Parkinson, J., Gerstein, M., Wodak, S., Emili, A., and Greenblatt, J. (2006). Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature*, 440(7084):637–643.

[96] Kuchaiev, O., Rašajski, M., Higham, D., and Pržulj, N. (2009). Geometric de-noising of protein-protein interaction networks. *PLoS Comp Biol*, 5(8):e1000454.

[97] Kufareva, I., Budagyan, L., Raush, E., Totrov, M., and Abagyan, R. (2007). Pier: protein interface recognition for structural proteomics. *Proteins: Structure, Function, and Bioinformatics*, 67(2):400–417.

[98] Kühlbrandt, W. (2014). Microscopy: cryo-em enters a new era. *Elife*, 3:e03678.

[99] Lage, K., Karlberg, E. O., Størling, Z. M., Olason, P. I., Pedersen, A. G., Rigina, O., Hinsby, A. M., Tümer, Z., Pociot, F., Tommerup, N., et al. (2007). A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nature biotechnology*, 25(3):309–316.

[100] Lambert, S. A., Jolma, A., Campitelli, L. F., Das, P. K., Yin, Y., Albu, M., Chen, X., Taipale, J., Hughes, T. R., and Weirauch, M. T. (2018). The human transcription factors. *Cell*, 172(4):650–665.

[101] Lancichinetti, A., Radicchi, F., and Ramasco, J. J. (2010). Statistical significance of communities in networks. *Physical Review E*, 81(4):046110.

[102] Landrum, M. J., Lee, J. M., Benson, M., Brown, G., Chao, C., Chitipiralla, S., Gu, B., Hart, J., Hoffman, D., Hoover, J., et al. (2015). Clinvar: public archive of interpretations of clinically relevant variants. *Nucleic acids research*, 44(D1):D862–D868.

[103] Lee, I., Blom, U. M., Wang, P. I., Shim, J. E., and Marcotte, E. M. (2011). Prioritizing candidate disease genes by network-based boosting of genome-wide association data. *Genome research*, pages gr–118992.

[104] Leoyklang, P., Suphapeetiporn, K., Srichomthong, C., Tongkobpetch, S., Fietze, S., Dorward, H., Cullinane, A. R., Gahl, W. A., Huizing, M., and Shotelersuk, V. (2013). Disorders with similar clinical phenotypes reveal underlying genetic interaction: Satb2 acts as an activator of the upf3b gene. *Human genetics*, 132(12):1383–1393.

[105] Li, J. and Lu, Z. (2013). Pathway-based drug repositioning using causal inference. *BMC bioinformatics*, 14(16):S3.

[106] Li, J., Zheng, S., Chen, B., Butte, A. J., Swamidass, S. J., and Lu, Z. (2015). A survey of current trends in computational drug repositioning. *Briefings in bioinformatics*, 17(1):2–12.

[107] Li, Y. and Patra, J. C. (2010). Genome-wide inferring gene–phenotype relationship by walking on the heterogeneous network. *Bioinformatics*, 26(9):1219–1224.

[108] Liang, S., Zhang, C., Liu, S., and Zhou, Y. (2006). Protein binding site prediction using an empirical scoring function. *Nucleic acids research*, 34(13):3698–3707.

[109] Lim, Y. H., Ovejero, D., Sugarman, J. S., DeKlotz, C. M., Maruri, A., Eichenfield, L. F., Kelley, P. K., Jüppner, H., Gottschalk, M., Tifft, C. J., et al. (2013). Multilineage somatic activating mutations in hras and nras cause mosaic cutaneous and skeletal lesions, elevated fgf23 and hypophosphatemia. *Human molecular genetics*, 23(2):397–407.

[110] Lin, D. et al. (1998). An information-theoretic definition of similarity. *Icml*, 98(1998):296–304.

[111] Liu, D., Ghosh, D., and Lin, X. (2008). Estimation and testing for the effect of a genetic pathway on a disease outcome using logistic kernel machine regression via logistic mixed models. *BMC bioinformatics*, 9(1):292.

[112] Liu, G., Wong, L., and Chua, H. N. (2009). Complex discovery from weighted ppi networks. *Bioinformatics*, 25(15):1891–1897.

[113] Liu, Y., Tennant, D. A., Zhu, Z., Heath, J. K., Yao, X., and He, S. (2014). Dime: a scalable disease module identification algorithm with application to glioma progression. *PloS one*, 9(2):e86693.

[114] Lockless, S. W. and Ranganathan, R. (1999). Evolutionarily conserved pathways of energetic connectivity in protein families. *Science*, 286(5438):295–299.

[115] Lu, L., Xia, Y., Paccanaro, A., Yu, H., and Gerstein, M. (2005). Assessing the limits of genomic data integration for predicting protein networks. *Genome Res*, 15(7):945–53.

[116] Magger, O., Waldman, Y. Y., Ruppin, E., and Sharan, R. (2012). Enhancing the prioritization of disease-causing genes through tissue specific protein interaction networks. *PLoS computational biology*, 8(9):e1002690.

[117] Maglott, D., Ostell, J., Pruitt, K. D., and Tatusova, T. (2006). Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 35(suppl_1):D26–D31.

[118] Mahdavi, M. A. and Lin, Y.-H. (2007). False positive reduction in protein-protein interaction predictions using Gene Ontology annotations. *BMC Bioinformatics*, 8:262.

[119] Marbach, D., Costello, J. C., Küffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., Allison, K. R., Aderhold, A., Bonneau, R., Chen, Y., et al. (2012). Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8):796.

[120] Marchegiani, S., Davis, T., Tessadori, F., Van Haaften, G., Brancati, F., Hoischen, A., Huang, H., Valkanas, E., Pusey, B., Schanze, D., et al. (2015). Recurrent mutations in the basic domain of twist2 cause ablepharon macrostomia and barber-say syndromes. *The American Journal of Human Genetics*, 97(1):99–110.

[121] Massa, M. S., Chiogna, M., and Romualdi, C. (2010). Gene set analysis exploiting the topology of a pathway. *BMC Systems Biology*, 4(1):1.

[122] Mathur, S. and Dinakarpandian, D. (2012). Finding disease similarity based on implicit semantic similarity. *Journal of biomedical informatics*, 45(2):363–371.

[123] Matthews, L. R., Vaglio, P., Reboul, J., Ge, H., Davis, B. P., Garrels, J., Vincent, S., and Vidal, M. (2001). Identification of potential interaction networks using sequence-based searches for conserved protein-protein interactions or "interologs". *Genome research*, 11(12):2120–2126.

[124] Matys, V., Kel-Margoulis, O. V., Fricke, E., Liebich, I., Land, S., Barre-Dirrie, A., Reuter, I., Chekmenev, D., Krull, M., Hornischer, K., et al. (2006). Transfac® and its module transcompel®: transcriptional gene regulation in eukaryotes. *Nucleic acids research*, 34(suppl_1):D108–D110.

[125] McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University (Baltimore, M. (2018). Online Mendelian Inheritance in Man, OMIM. http://omim.org.

[126] McPherson, E. and Clemens, M. (1997). Bruck syndrome (osteogenesis imperfecta with congenital joint contractures): review and report on the first north american case. *American journal of medical genetics*, 70(1):28–31.

[127] Menche, J., Sharma, A., Kitsak, M., Ghiassian, S. D., Vidal, M., Loscalzo, J., and Barabási, A.-L. (2015). Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347(6224):1257601.

[128] Menden, M. P., Iorio, F., Garnett, M., McDermott, U., Benes, C. H., Ballester, P. J., and Saez-Rodriguez, J. (2013). Machine learning prediction of cancer cell sensitivity to drugs based on genomic and chemical properties. *PLoS one*, 8(4):e61318.

[129] Metzker, M. L. (2010). Sequencing technologies—the next generation. *Nature reviews genetics*, 11(1):31.

[130] Meyer, M. J., Beltrán, J. F., Liang, S., Fragoza, R., Rumack, A., Liang, J., Wei, X., and Yu, H. (2018). Interactome insider: a structural interactome browser for genomic studies. *Nature methods*, 15(2):107.

[131] Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. (2013). *Machine learning: An artificial intelligence approach*. Springer Science & Business Media.

[132] Milenković, T. and Pržulj, N. (2008). Uncovering biological network function via graphlet degree signatures. *Cancer informatics*, 6:CIN–S680.

[133] Milo, R. (2013). What is the total number of protein molecules per cell volume? a call to rethink some published values. *BioEssays*, 35(12):1050–1055.

[134] Morcos, F., Pagnani, A., Lunt, B., Bertolino, A., Marks, D. S., Sander, C., Zecchina, R., Onuchic, J. N., Hwa, T., and Weigt, M. (2011). Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301.

[135] Mordelet, F. and Vert, J.-P. (2011). Prodige: Prioritization of disease genes with multitask machine learning from positive and unlabeled examples. *BMC bioinformatics*, 12(1):389.

[136] Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C., and Morris, Q. (2008). Genemania: a real-time multiple association network integration algorithm for predicting gene function. *Genome biology*, 9(1):S4.

[137] Mullard, A. (2014). New drugs cost us $2.6 billion to develop.

[138] Napolitano, F., Zhao, Y., Moreira, V. M., Tagliaferri, R., Kere, J., D'Amato, M., and Greco, D. (2013). Drug repositioning: a machine-learning approach through data integration. *Journal of cheminformatics*, 5(1):30.

[139] Navlakha, S. and Kingsford, C. (2010). The power of protein interaction networks for associating genes with diseases. *Bioinformatics*, 26(8):1057–1063.

[140] Nepusz, T., Yu, H., and Paccanaro, A. (2012). Detecting overlapping protein complexes in protein-protein interaction networks. *Nature methods*, 9(5):471.

[141] Ober, C. and Hoffjan, S. (2006). Asthma genetics 2006: the long and winding road to gene discovery. *Genes and immunity*, 7(2):95.

[142] of Health. National Human Genome Research Institute., N. I. (2018). Talking glossary of genetic terms. https://www.genome.gov/glossary.

[143] of Medicine, U. S. N. L. (2018). The Medical Subject Headings (MeSH). https://www.nlm.nih.gov/mesh/.

[144] Ogris, C., Guala, D., Kaduk, M., and Sonnhammer, E. L. (2017). Funcoup 4: new species, data, and visualization. *Nucleic acids research*, 46(D1):D601–D607.

[145] Oliver, S. (2000). Proteomics: guilt-by-association goes global. *Nature*, 403(6770):601.

[146] Organization, W. H. et al. (1996). Guidelines for atc classification and ddd assignment. In *Guidelines for ATC classification and DDD assignment*. World Health Organization.

[147] Organization, W. H. et al. (2006). International classification of diseases (icd). *http://www. who. int/classifications/icd/en/*.

[148] Oti, M., Snel, B., Huynen, M. A., and Brunner, H. G. (2006). Predicting disease genes using protein–protein interactions. *Journal of medical genetics*, 43(8):691–698.

[149] Ott, J. (1999). *Analysis of human genetic linkage*. JHU Press.

[150] Panepinto, J. A., Torres, S., Bendo, C. B., McCavit, T. L., Dinu, B., Sherman-Bien, S., Bemrich-Stolz, C., and Varni, J. W. (2013). Pedsql™ sickle cell disease module: feasibility, reliability, and validity. *Pediatric blood & cancer*, 60(8):1338–1344.

[151] Park, S., Yang, J.-S., Shin, Y.-E., Park, J., Jang, S. K., and Kim, S. (2011). Protein localization as a principal feature of the etiology and comorbidity of genetic diseases. *Molecular systems biology*, 7(1):494.

[152] Pawson, T. (1995). Protein modules and signalling networks. *Nature*, 373(6515):573.

[153] Penrose, M. D. (2003). *Random geometric graphs*, volume 5 of *Oxford Studies in Probability*. Oxford University Press.

[154] Pesquita, C., Faria, D., Bastos, H., Ferreira, A. E., Falcão, A. O., and Couto, F. M. (2008). Metrics for go based protein semantic similarity: a systematic evaluation. *BMC bioinformatics*, 9(5):S4.

[155] Petajan, J. H., Momberger, G. L., Aase, J., and Wright, D. G. (1969). Arthrogryposis syndrome (kuskokwim disease) in the eskimo. *Jama*, 209(10):1481–1486.

[156] Pieper, U., Eswar, N., Davis, F. P., Braberg, H., Madhusudhan, M. S., Rossi, A., Marti-Renom, M., Karchin, R., Webb, B. M., Eramian, D., et al. (2006). Modbase: a database of annotated comparative protein structure models and associated resources. *Nucleic acids research*, 34(suppl_1):D291–D295.

[157] Pierce, B. G., Hourai, Y., and Weng, Z. (2011). Accelerating protein docking in zdock using an advanced 3d convolution library. *PloS one*, 6(9):e24657.

[158] Piñero, J., Bravo, À., Queralt-Rosinach, N., Gutiérrez-Sacristán, A., Deu-Pons, J., Centeno, E., García-García, J., Sanz, F., and Furlong, L. I. (2016). Disgenet: a comprehensive platform integrating information on human disease-associated genes and variants. *Nucleic acids research*, page gkw943.

[159] Porollo, A. and Meller, J. (2007). Prediction-based fingerprints of protein–protein interactions. *Proteins: Structure, Function, and Bioinformatics*, 66(3):630–645.

[160] Prasad, T. K., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., Telikicherla, D., Raju, R., Shafreen, B., Venugopal, A., et al. (2009). Human protein reference database - 2009 update. *Nucleic acids research*, 37(suppl 1):D767–D772.

[161] Pržulj, N., Corneil, D., and Jurisica, I. (2004). Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–15.

[162] Pržulj, N. and Higham, D. (2006). Modelling protein-protein interaction networks via a stickiness index. *J Roy Soc Interface*, 3(10):711–6.

[163] Pržulj, N. (2007). Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183.

[164] Radivojac, P., Clark, W. T., Oron, T. R., Schnoes, A. M., Wittkop, T., Sokolov, A., Graim, K., Funk, C., Verspoor, K., Ben-Hur, A., et al. (2013). A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221.

[165] Radivojac, P., Peng, K., Clark, W. T., Peters, B. J., Mohan, A., Boyle, S. M., and Mooney, S. D. (2008). An integrated approach to inferring gene–disease associations in humans. *Proteins: Structure, Function, and Bioinformatics*, 72(3):1030–1037.

[166] Ramasamy, A., Trabzuni, D., Guelfi, S., Varghese, V., Smith, C., Walker, R., De, T., Hardy, J., Ryten, M., Weale, M. E., et al. (2014). Genetic variability in the regulation of gene expression in ten regions of the human brain. *Nature neuroscience*, 17(10):1418.

[167] Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of artificial intelligence research*, 11:95–130.

[168] Reuter, J. A., Spacek, D. V., and Snyder, M. P. (2015). High-throughput sequencing technologies. *Molecular cell*, 58(4):586–597.

[169] Rogers, D., Brown, R. D., and Hahn, M. (2005). Using extended-connectivity fingerprints with laplacian-modified bayesian analysis in high-throughput screening follow-up. *Journal of biomolecular screening*, 10(7):682–686.

[170] Rosenfeld, J. A., Ballif, B. C., Lucas, A., Spence, E. J., Powell, C., Aylsworth, A. S., Torchia, B. A., and Shaffer, L. G. (2009). Small deletions of satb2 cause some of the clinical features of the 2q33. 1 microdeletion syndrome. *PloS one*, 4(8):e6568.

[171] Rual, J., Venkatesan, K., Hao, T., Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G., Gibbons, F., Dreze, M., Ayivi-Guedehoussou, N., Klitgord, N., Simon, C., Boxem, M., Milstein, S., Rosenberg, J., Goldberg, D., Zhang, L., Wong, S., Franklin, G., Li, S., Albala, J., Lim, J., Fraughton, C., Llamosas, E., Cevik, S., Bex, C., Lamesch, P., Sikorski, R., Vandenhaute, J., Zoghbi, H., Smolyar, A., Bosak, S., Sequerra, R., Doucette-Stamm, L., Cusick, M., Hill, D., Roth, F., and Vidal, M. (2005). Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437(7062):1173–8.

[172] Salton, G. and McGill, M. J. (1986). *Introduction to modern information retrieval*. McGraw-Hill, Inc.

[173] Saxena, R., Voight, B. F., Lyssenko, V., Burtt, N. P., de Bakker, P. I., Chen, H., Roix, J. J., Kathiresan, S., Hirschhorn, J. N., Daly, M. J., et al. (2007). Genome-wide association analysis identifies loci for type 2 diabetes and triglyceride levels. *Science*, 316(5829):1331–1336.

[174] Schadt, E. E. (2009). Molecular networks as sensors and drivers of common human diseases. *Nature*, 461(7261):218.

[175] Schaefer, M. H., Fontaine, J.-F., Vinayagam, A., Porras, P., Wanker, E. E., and Andrade-Navarro, M. A. (2012). HIPPIE: Integrating protein interaction networks with experiment based quality scores. *PloS one*, 7(2):e31826.

[176] Schenone, M., Dančík, V., Wagner, B. K., and Clemons, P. A. (2013). Target identification and mechanism of action in chemical biology and drug discovery. *Nature chemical biology*, 9(4):232.

[177] Schloss, J. A. (2008). How to get genomes at one ten-thousandth the cost. *Nature biotechnology*, 26(10):1113.

[178] Schmitt, T., Ogris, C., and Sonnhammer, E. L. (2013). Funcoup 3.0: database of genome-wide functional coupling networks. *Nucleic acids research*, 42(D1):D380–D388.

[179] Schriml, L. M., Arze, C., Nadendla, S., Chang, Y.-W. W., Mazaitis, M., Felix, V., Feng, G., and Kibbe, W. A. (2011). Disease ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946.

[180] Seco, N., Veale, T., and Hayes, J. (2004). An intrinsic information content metric for semantic similarity in wordnet. In *ECAI*, volume 16, page 1089.

[181] Shaheen, R., Al-Owain, M., Sakati, N., Alzayed, Z. S., and Alkuraya, F. S. (2010). Fkbp10 and bruck syndrome: phenotypic heterogeneity or call for reclassification? *The American Journal of Human Genetics*, 87(2):306–307.

[182] Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.

[183] Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55.

[184] Sharan, R., Ulitsky, I., and Shamir, R. (2007). Network-based prediction of protein function. *Molecular systems biology*, 3(1):88.

[185] Shi, C., Kong, X., Huang, Y., Philip, S. Y., and Wu, B. (2014). Hetesim: A general framework for relevance measure in heterogeneous networks. *IEEE Trans. Knowl. Data Eng.*, 26(10):2479–2492.

[186] Singh-Blom, U. M., Natarajan, N., Tewari, A., Woods, J. O., Dhillon, I. S., and Marcotte, E. M. (2013). Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PloS one*, 8(5):e58977.

[187] Slater, G. S. C. and Birney, E. (2005). Automated generation of heuristics for biological sequence comparison. *BMC bioinformatics*, 6(1):31.

[188] Smith, C. L., Goldsmith, C.-A. W., and Eppig, J. T. (2005). The mammalian phenotype ontology as a tool for annotating, analyzing and comparing phenotypic information. *Genome biology*, 6(1):R7.

[189] Smyth, M. and Martin, J. (2000). x ray crystallography. *Molecular Pathology*, 53(1):8.

[190] Srihari, S., Yong, C. H., Patil, A., and Wong, L. (2015). Methods for protein complex prediction and their contributions towards understanding the organisation, function and dynamics of complexes. *FEBS letters*, 589(19):2590–2602.

[191] Steinwart, I. and Christmann, A. (2008). *Support vector machines*. Springer Science & Business Media.

[192] Stelzl, U., Worm, U., Lalowski, M., Haenig, C., Brembeck, F., Goehler, H., Stroedicke, M., Zenkner, M., Schoenherr, A., Koeppen, S., Timm, J., Mintzlaff, S., Abraham, C., Bock, N., Kietzmann, S., Goedde, A., Toksöz, E., Droege, A., Krobitsch, S., Korn, B., Birchmeier, W., Lehrach, H., and Wanker, E. (2005). A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–68.

[193] Stenson, P. D., Ball, E. V., Mort, M., Phillips, A. D., Shaw, K., and Cooper, D. N. (2012). The human gene mutation database (hgmd) and its exploitation in the fields of personalized genomics and molecular evolution. *Current protocols in bioinformatics*, 39(1):1–13.

[194] Stuart, J. M., Segal, E., Koller, D., and Kim, S. K. (2003). A gene-coexpression network for global discovery of conserved genetic modules. *science*, 302(5643):249–255.

[195] Szklarczyk, D., Morris, J. H., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N. T., Roth, A., Bork, P., et al. (2016). The string database in 2017: quality-controlled protein–protein association networks, made broadly accessible. *Nucleic acids research*, page gkw937.

[196] Tan-Sindhunata, M. B., Mathijssen, I. B., Smit, M., Baas, F., De Vries, J. I., Van Der Voorn, J. P., Kluijt, I., Hagen, M. A., Blom, E. W., Sistermans, E., et al. (2015). Identification of a dutch founder mutation in musk causing fetal akinesia deformation sequence. *European Journal of Human Genetics*, 23(9):1151.

[197] Taylor, I. W., Linding, R., Warde-Farley, D., Liu, Y., Pesquita, C., Faria, D., Bull, S., Pawson, T., Morris, Q., and Wrana, J. L. (2009). Dynamic modularity in protein interaction networks predicts breast cancer outcome. *Nature biotechnology*, 27(2):199.

[198] Torres, M., Cáceres, J. J., Jiménez, R., Yubero, V., Vega, C., Rolón, M., Cernuzzi, L., Barán, B., and Paccanaro, A. (2017). Drug cocktail selection for the treatment of chagas disease: A multi-objective approach. In *Computer Conference (CLEI), 2017 XLIII Latin American*, pages 1–5. IEEE.

[199] Tranchevent, L.-C., Barriot, R., Yu, S., Van Vooren, S., Van Loo, P., Coessens, B., De Moor, B., Aerts, S., and Moreau, Y. (2008). Endeavour update: a web resource for gene prioritization in multiple species. *Nucleic acids research*, 36(suppl_2):W377–W384.

[200] Trapnell, C., Williams, B. A., Pertea, G., Mortazavi, A., Kwan, G., Van Baren, M. J., Salzberg, S. L., Wold, B. J., and Pachter, L. (2010). Transcript assembly and quantification by rna-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature biotechnology*, 28(5):511.

[201] Tropsha, A., Gramatica, P., and Gombar, V. K. (2003). The importance of being earnest: validation is the absolute essential for successful application and interpretation of qspr models. *QSAR & Combinatorial Science*, 22(1):69–77.

[202] Uetz, P., Giot, L., Cagney, G., Mansfield, T., Judson, R., Knight, J., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M.,

Fields, S., and Rothberg, J. (2000). A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. *Nature*, 403(6770):623–7.

[203] Van Buggenhout, G., Van Ravenswaaij-Arts, C., Maas, N. M., Thoelen, R., Vogels, A., Smeets, D., Salden, I., Matthijs, G., Fryns, J.-P., and Vermeesch, J. (2005). The del (2)(q32. 2q33) deletion syndrome defined by clinical and molecular characterization of four patients. *European journal of medical genetics*, 48(3):276–289.

[204] Van Dongen, S. M. (2000). *Graph clustering by flow simulation*. PhD thesis, Utrecht University.

[205] Van Driel, M. A., Bruggeman, J., Vriend, G., Brunner, H. G., and Leunissen, J. A. (2006). A text-mining analysis of the human phenome. *European journal of human genetics*, 14(5):535–542.

[206] Vanunu, O., Magger, O., Ruppin, E., Shlomi, T., and Sharan, R. (2010). Associating genes and protein complexes with disease via network propagation. *PLoS Comput Biol*, 6(1):e1000641.

[207] Vanunu, O. and Sharan, R. (2008). A propagation-based algorithm for inferring gene-disease assocations. In *German Conference on Bioinformatics*, pages 54–52.

[208] Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., et al. (2001). The sequence of the human genome. *science*, 291(5507):1304–1351.

[209] Visscher, P. M., Brown, M. A., McCarthy, M. I., and Yang, J. (2012). Five years of gwas discovery. *The American Journal of Human Genetics*, 90(1):7–24.

[210] Visscher, P. M., Wray, N. R., Zhang, Q., Sklar, P., McCarthy, M. I., Brown, M. A., and Yang, J. (2017). 10 years of gwas discovery: biology, function, and translation. *The American Journal of Human Genetics*, 101(1):5–22.

[211] Wang, H., Kakaradov, B., Collins, S. R., Karotki, L., Fiedler, D., Shales, M., Shokat, K. M., Walther, T. C., Krogan, N. J., and Koller, D. (2009). A complex-based reconstruction of the saccharomyces cerevisiae interactome. *Molecular & Cellular Proteomics*, 8(6):1361–1381.

[212] Wang, X., Dalkic, E., Wu, M., and Chan, C. (2008). Gene module level analysis: identification to networks and dynamics. *Current opinion in biotechnology*, 19(5):482–491.

[213] Wang, X., Gulbahce, N., and Yu, H. (2011). Network-based methods for human disease gene prediction. *Briefings in functional genomics*, 10(5):280–293.

[214] Wang, Y., Chen, S., Deng, N., and Wang, Y. (2013). Drug repositioning by kernel-based integration of molecular structure, molecular activity, and phenotype data. *PloS one*, 8(11):e78518.

[215] Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *nature*, 393(6684):440.

[216] Webster, P., Dawes, J. C., Dewchand, H., Takacs, K., Iadarola, B., Bolt, B. J., Caceres, J. J., Kaczor, J., Dharmalingam, G., Dore, M., et al. (2018). Subclonal mutation selection in mouse lymphomagenesis identifies known cancer loci and suggests novel candidates. *Nature communications*, 9(1):2649.

[217] Wilkie, A. (1994). The molecular basis of genetic dominance. *Journal of medical genetics*, 31(2):89–98.

[218] Wilson, R. and Watkins, J. J. (2013). *Combinatorics: ancient & modern*. OUP Oxford.

[219] Wishart, D. S., Knox, C., Guo, A. C., Cheng, D., Shrivastava, S., Tzur, D., Gautam, B., and Hassanali, M. (2008). DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic acids research*, 36(suppl 1):D901–D906.

[220] Wishart, D. S., Tzur, D., Knox, C., Eisner, R., Guo, A. C., Young, N., Cheng, D., Jewell, K., Arndt, D., Sawhney, S., et al. (2007). HMDB: the human metabolome database. *Nucleic acids research*, 35(suppl 1):D521–D526.

[221] Wu, C., Gudivada, R. C., Aronow, B. J., and Jegga, A. G. (2013). Computational drug repositioning through heterogeneous network clustering. *BMC systems biology*, 7(5):S6.

[222] Wu, X., Jiang, R., Zhang, M. Q., and Li, S. (2008). Network-based global inference of human disease genes. *Molecular systems biology*, 4(1):189.

[223] Xie, M., Xu, Y., Zhang, Y., Hwang, T., and Kuang, R. (2015). Network-based phenome-genome association prediction by bi-random walk. *PloS one*, 10(5):e0125138.

[224] Xu, Y., Jiang, H., Tyler-Smith, C., Xue, Y., Jiang, T., Wang, J., Wu, M., Liu, X., Tian, G., Wang, J., et al. (2011). Comprehensive comparison of three commercial human whole-exome capture platforms. *Genome biology*, 12(9):R95.

[225] Yang, H., Nepusz, T., and Paccanaro, A. (2012). Improving go semantic similarity measures by exploring the ontology beneath the terms and modelling uncertainty. *Bioinformatics*, 28(10):1383–1389.

[226] Yates, B., Braschi, B., Gray, K. A., Seal, R. L., Tweedie, S., and Bruford, E. A. (2016). Genenames. org: the hgnc and vgnc resources in 2017. *Nucleic acids research*, page gkw1033.

[227] Yıldırım, M. A., Goh, K.-I., Cusick, M. E., Barabási, A.-L., and Vidal, M. (2007). Drug—target network. *Nature biotechnology*, 25(10):1119.

[228] Yip, K. Y., Cheng, C., and Gerstein, M. (2013). Machine learning and genome annotation: a match meant to be? *Genome biology*, 14(5):205.

[229] Yu, H., Paccanaro, A., Trifonov, V., and Gerstein, M. (2006). Predicting interactions in protein networks by completing defective cliques. *Bioinformatics*, 22(7):823–9.

[230] Zdobnov, E. M. and Apweiler, R. (2001). Interproscan–an integration platform for the signature-recognition methods in interpro. *Bioinformatics*, 17(9):847–848.

[231] Zeng, X., Liao, Y., Liu, Y., and Zou, Q. (2017). Prediction and validation of disease genes using hetesim scores. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 14(3):687–695.

[232] Zenteno, J. C., Crespí, J., Buentello-Volante, B., Buil, J. A., Bassaganyas, F., Vela-Segarra, J. I., Diaz-Cascajosa, J., and Marieges, M. T. (2014). Next generation sequencing uncovers a missense mutation in col4a1 as the cause of familial retinal arteriolar tortuosity. *Graefe's Archive for Clinical and Experimental Ophthalmology*, 252(11):1789–1794.

[233] Zhang, A. (2009). *Protein interaction networks: computational analysis*. Cambridge University Press.

[234] Zhang, P., Wang, F., and Hu, J. (2014). Towards drug repositioning: a unified computational framework for integrating multiple aspects of drug similarity and disease similarity. In *AMIA Annual Symposium Proceedings*, volume 2014, page 1258. American Medical Informatics Association.

[235] Zhang, Z., Zhao, M., and Chow, T. W. (2015). Graph based constrained semi-supervised learning framework via label propagation over adaptive neighborhood. *IEEE Transactions on Knowledge and Data Engineering*, 27(9):2362–2376.

[236] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. (2004). Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*, pages 321–328. MIT Press.

[237] Zhu, X. (2006). Semi-supervised learning literature survey. *Computer Science, University of Wisconsin-Madison*, 2(3):4.

[238] Zhu, X., Gerstein, M., and Snyder, M. (2007). Getting connected: analysis and principles of biological networks. *Genes & development*, 21(9):1010–1024.